Putting data delivery into context: Design and evaluation of adaptive networking support for successful communication in wireless self-organizing networks

Aline Carneiro Viana

HABILITATION À DIRIGER DES RECHERCHES

Putting data delivery into context:

Design and evaluation of adaptive networking support for successful communication in wireless self-organizing networks

Aline Carneiro Viana

Submitted in total fulfilment of the requirements of the degree of Habilitation à Diriger des Recherches

Defense on December 14, 2011

Committee:

Reviewers:

Catherine Rosenberg	Prof.	University of Warterloo
David Simplot-Ryl	Prof.	Université Lille1
Andrzej Duda	Prof.	Grenoble INP-Ensimag

Examiners:

Patrick Thiran	Prof.	École Polytechnique Fédérale de Lausanne
Roy Friedman	Prof.	Technion
Fabrice Valois	Prof.	INSA Lyon
Serge Fdida	Prof.	UPMC – Sorbonne Universités

À toutes les personnes importantes dans ma vie.
En spécial à mon époux et à ma précieuse famille.

Remerciements

Je tiens tout d'abord à remercier chaleureusement Catherine Rosenberg, David Simplot-Ryl et Andrzej Duda qui m'ont fait l'honneur de rapporter sur mon manuscrit avant la soutenance. Je tiens également à exprimer ma gratitude envers Patrick Thiran, Roy Friedman et Fabrice Valois qui ont accepté de participer à mon jury en tant que examinateurs. En fin, je tiens à remercier Serge Fdida, Luigi Iannone, Marcelo Dias de Amorim, Fatiha Zaidi et Hakima Chaouchi pour leur soutien constant et amical tout au long du processus de préparation de cette habilitation. Qu'ils soient tous assurés de ma plus profonde reconnaissance pour l'attention qu'ils ont portée à ce manuscrit et pour le temps précieux qu'ils m'ont consacré. Je suis honorée de pouvoir leur présenter mes travaux.

Je souhaite remercier Anne-Marie Kermarrec qui m'a fait confiance en 2006 en m'intégrant à son équipe projet INRIA ASAP comme Chargée de Recherche. Pendant ces premières années, elle m'a guidé dans mes choix et a su retenir ma vivacité naturelle lorsque cela devenait nécessaire sans pour cela que je ne ressente ni contraintes, ni entraves, ni obligations.

Je tiens à exprimer toute ma sympathie à Adam Wolisz et aux membres de son groupe TKN qui m'ont chaleureusement accueilli durant mon année sabbatique passée à TU-Berlin, en Allemagne. Je tiens aussi à exprimer toute ma gratitude à Philippe Jacquet qui a accepté de m'intégrer à son équipe projet INRIA Hipercom, après mon retour d'Allemagne.

Je tiens à remercier toutes les différentes personnes qui m'ont inspirées depuis la fin de ma thèse et grâce à qui j'ai pu mener à bien mes recherches. J'ai énormément appris sur le métier de chercheur à leur contact et si je me sens aujourd'hui capable d'encadrer des recherches, c'est grâce à leur exemple. La peur d'oublier des noms m'oblige à m'abstenir de les citer, mais je suis convaincue que vous vous reconnaîtrez.

J'ai eu la chance et le bonheur de travailler, depuis le début de ma carrière, avec de nombreuses personnes brillantes. Ainsi, les travaux de recherche présentés dans ce document ont été le fruit de nombreuses collaborations et plusieurs chercheurs y ont contribués de manière significative. Je remercie enfin chaleureusement tous ces nombreux collègues et amis ainsi que tous les étudiants que j'ai encadrés et qui ont contribué aux travaux présentés dans ce mémoire.

Une profonde reconnaissance va à mes parents, Sonia Maria Carneiro Viana et Archimedes Viana, pour m'avoir transmis des valeurs essentielles et qui m'ont soutenu au mieux dans la poursuite de mes études jusqu'au doctorat. Je pense aussi à mes deux frères Alan Carneiro Viana et Archimedes Viana Junior pour leur décalage total du milieu universitaire, qui m'aide à garder les pieds sur terre, pour leur sens de la contemplation et de la rêverie, et pour leur grand sens de l'humour. Un grand merci aussi à toute ma belle-famille pour les moments chaleureux que nous pouvons partager ensemble. Merci aussi à tous mes amis pour leur affection et soutien.

Enfin, un grand merci, plein d'affection, à mon époux Luigi Iannone pour son soutien sans faille dans mes choix, pour ses rires, son sens de l'humour, ses blagues parfois décalées, ses conseils, sa simplicité, ses attentions, son respect professionnel, son honnêteté, sa curiosité, sa confiance, sa patience, sa compréhension, son écoute, sa présence et enfin, pour tout le bonheur qu'il me procure au quotidien.

Abstract

This document is dedicated to my research work developed during the latest 6 years on the design and evaluation of wireless networking systems and is the result of a number of collaborations. In particular, my main goal has been the provision of networking support for success data delivery in wireless self-organizing networks. The central question that has been driving my research activities is: *"what are the networking services underlying the design of successful communication strategies in wireless self-organizing networking systems (static or mobile)?"*.

Wireless self-organizing networks (WSON) have intrinsic characteristics and consequently, require particular solutions that set them apart from traditional "graph-based" networks. The different types of WSONs require adaptive networking services targeted to deal with their dynamic nature (i.e., mobility, resource limitation, unreliable wireless communication, etc) and to find a fit between their operation and the environment. Influenced by such observations, my research activities were guided by the main goal of providing *network-level support for success data delivery in wireless self-organizing networks*. The research axes I developed together with my colleagues in this context are categorized in adaptive core and network-level services. These two categories of services are distinguished by the level where adaptation is considered, i.e., at the node or at the network level, respectively. The contributions related to core services that I performed relate to location and neighborhood discovery services. Due to page limitation, this manuscript is, however, devoted to the research that I conducted around *adaptive network-level services*. Therefore, it is structured in three main chapters corresponding to three classes of network-level services: topology management services, data management services, and routing and forwarding services.

My first presented research contributions concern topology management services performed through node adaptation – by imposing a hierarchy in the network through clustering or by removing nodes from the network graph by powering them off – and through controlled mobility – which affects both the presence of nodes and links, as well as the quality of links in the network graph. Related to node adaptation, the *SAND* protocol and the systems *VINCOS* and *NetGeoS* dealing respectively with energy-conserving topology management and with geometric self-structuring in wireless sensor networks (WSNs) were proposed. Then, related to controlled mobility, the Hilbert-based trajectory design and *Cover* approaches are introduced. They focus on the deployment of solutions for zone coverage with mobile nodes, designed to periodically monitor a geographic area or to cover mobile sensor nodes (targets). Considering data management services, my contributions relate to data collection – which involved data distribution solutions with organization goals – and data dissemination –where data flows are directed towards the network. For this, the protocols *DEEP* and *Supple* were designed for wireless sensor networks, while *FairMix* and *VIP delegation* approaches focus on information dissemination in wireless social networks. In particular, to improve data dissemination, *FairMix* and *VIP delegation* exploit social interests' similarities of people or groups in static networks or the social aspect of their wireless interactions in mobile networks. Finally, my works on adaptive forwarding services address connectivity opportunity in delay tolerant networks. In this context, *Seeker* and *GrAnt* protocols were designed and use respectively contact history (contact and communication patterns) and social network properties of nodes to predict future meetings and to better adjust forwarding decisions.

Following the new communication opportunities and the dynamic shift observed over the past years in wireless networks, my research activities have been gradually moving from connected self-organizing networks to intermittently connected and opportunistic networks. In this way, my future research focus on: (1) leveraging the uncontrolled mobility patterns of pervasive mobile sensing devices to improve sensing collaborative efforts; (2) looking deeper into social graph generation techniques from contact traces; (3) studying what are the factors impacting (in a positive or in a negative way) the success of information dissemination in social mobile networks; and (4) investigating the possibility of tailoring network coding for information dissemination in social mobile networks.

Contents

1

Introduction

1.1 Foreword

This document is dedicated to my research work developed during the latest 6 years on the design and evaluation of wireless networking systems. In particular, my main goal has been the provision of networking support for success data delivery in wireless self-organizing networks. The central question that has been driving my research activities is: *"what are the networking services underlying the design of successful communication strategies in wireless self-organizing networking systems (static or mobile)?"*.

Since the beginning of my carrier, I have the pleasure of working with collaborators with different experiences, professional positions, and from different locations. They are all brilliant and helped me progress as a researcher. By advising me, through the opportunity to advise them, or through collaborations, they contribute to my maturity as a researcher. For all this, I claim the provided synthesis and the achieved successes of my carrier work are also theirs. It is worth to mention this document contains then sentences and paragraphs of our common publications and technical reports.

In order to respect the page number limitation (40 pages) imposed by the Université Pierre et Marie Curie (UPMC) - Sorbonne Universités, this document (1) provides only an overview of part of the research I have been developing together with my collaborators, and (2) only mention a few references and those of my own articles. Having to select among my research activities the ones to be detailed here constituted then the main difficulty I had in preparing this document. For more technical details about the presented contributions and the detailed bibliographies, I invite the reader to refer to my publications mentioned in this document. At the end of this document, besides concluding it, I also present the perspectives for my further research, which I intent to develop in the next years.

My first steps to the research carrier in wireless networking systems started nine years ago and, since then, my activities have been performed in three different environments: two in France, at LIP6/UPMC (2002-2005, where I did my PhD) and at INRIA (from 2005-), and another in Germany, at the Telecommunication Networking group (TKN) of the Technical University of Berlin (2009-2010), where I spent one year working as invited researcher.

This document presents, however, the research work I developed after my PhD defense (July 2005). My earlier contributions have obviously influenced those presented in this document. Thus, in this section, I present an overview of my PhD research findings which drove my next research contributions and associated scientific approaches.

Since my PhD, my attention has focused on the proliferation of wireless computing and communication devices, coming with a multitude of applications, data, and services. Such diversity and a large scalability of the wireless communication landscape gave me the first motivations *to head to the area of self-organizing networking* and *to consider the challenges involved in guaranteeing successful data delivery in such networks.*

My PhD thesis work was performed at the LIP6 laboratory of the Université Pierre et Marie Curie (UPMC)– Sorbonne Universités from January 2002 to July 2005 and focused on the "Locating and routing in large scale self-organizing networks: from distributed hash tables to adaptive addressing structures" [1]. In particular, *scalability support* was one of the main design objectives of my PhD thesis. Providing a scalable and efficient data delivery service in the context of wireless ad hoc self-organizing systems is a difficult problem, due to the spontaneity of self-organizing networks. In response to this, during my PhD, I looked for an adaptive addressing structure and a mathematical space that eases location in a self-organized fashion and allows low management overhead. In order to provide scalability functionalities, I considered the integration of the Distributed Hash Tables (DHTs) abstraction in the network-level routing systems. The contributions of my thesis were the design of *Tribe* [2, 3, 4, 5] and *Twins* [6, 7, 8, 9] protocols. Tribe defines a tree-like mathematical structure to assign addresses to nodes, and to forward data, without any central control entity or positioning mechanism. Twins specifies a multidimensional Cartesian addressing structure, which is a strict mathematical representation of the network geographic space obtained through Hilbert space-filling curves. The geographic space is used for addressing and routing, while location is performed on the designed mathematical structure.

The different WSON types and involved applications may however, require other supports (a part from scalability) for guaranteeing successful data delivery, such as fast feedback, reliability, robustness, high automaticity, load balancing, to cite a few. Such requirement particularities have driven most of contributions that came later in my research career.

At the end of my PhD, the attention of the networking research community had turned to the so-called *"Internet of Things"*: a "wired" core interconnecting network "clouds" that gravitate at the edges. These clouds would be wired or wireless, infrastructure-based or not, deployed impromptu or in a planned way. The network community expectation was that the Next-Generation Internet would likely interconnect a much wider variety of devices and would also be highly heterogeneous in terms of the types of networks it interconnects.

Such expectation also relates to the pervasive computing age firstly envisaged by Mark Weiser in 1988 [10], in which individuals continually interact with hundreds of nearby wirelessly-interconnected computing devices. Although being correct in his prediction, computing devices at that period were not small enough or sufficiently spread to allow ubiquitous communication of users. Nevertheless, thanks to the combination of a variety of new emerging factors including (i) pervasiveness of computing devices with increasingly higher communication capabilities and smaller form factor (e.g., smartphones), (ii) ubiquitous wireless communication capability, and (iii) emergence of new applications, Mark Weiser's and the Next-Generation Internet's expectations are becoming a reality.

As a consequence, new emerging needs are appearing in terms of self-organization. In particular, once deployed, those networking clouds will need to spontaneously self-organize, forming a network interconnecting participating devices in environments where they might be exposed to a high degree of unpredictable variations, such as scarce resource, sparse connectivity, and unreliable links due to users' mobility or lack of network infrastructure. Hence, the research activities that followed my PhD were oriented to the provision of communication support in such new challenging environments.

In the following, I introduce the main context delimiting my research: *wireless self-organizing networks* and *adaptive services*. Then, the positioning of my research in this context and the organization of this document are presented at the end of this chapter.

1.2 Coping with network dynamics

(Publications [11, 12]: In collaboration with C. Sengul, A. Ziviani, S. Maag and F. Zaidi)

Although self-organization can be defined in many different ways, in the context of wireless networks, I use the definition describing a system that built-up progressively and is able (1) to provide flexibility and spontaneity in response to user requirements' and operating conditions' changes; (2) to configure and to reconfigure themselves automatically, as nodes appear, move, and disappear; and (3) to make it easier to maintain than ever. This result in wireless networks increasingly requiring intelligent reconfiguration ability to changing conditions; in other words, the ability to find a fit between their operation and the environment. Such adaptivity allows a network to operate more efficiently and predictably under a broader range of varying conditions. Consequently, adaptivity provides robustness and resilience. Therefore, self-organizing networks are expected to be more insensitive to perturbations or faults, and be able to recover fast from the effects of network dynamics.

The remainder of this section details different types of self-organizing networks and the involved network dynamics. We also present a categorization for adaptive services, classifying what type of services adapts to changes, when, and how frequently.

1.2.1 Wireless Self-Organizing Networks

My focus has been mainly on the idea of wireless self-organizing networks (WSONs) having the potential to build "networks anytime and anywhere" without any need for pre-existing infrastructure. The absence of fixed infrastructure means that the nodes communicate directly with one another in a peer-to-peer fashion and provide themselves the basic communication services, such as routing, topology and resource management, etc. WSON applications cover various areas, such as military or post-disaster rescue operations, area monitoring, event surveillance, group collaboration at conferences, campuses, or cities, etc. In this context, different types of WSONs – e.g., wireless ad hoc networks, wireless sensor networks, cognitive radio networks, and delay-tolerant networks (i.e., mobile networks with intermittent connectivity) – have a multitude of variable features, which can benefit from self-organization. In the following, I summarize the types of WSONs I have been considering after my PhD:

- *Wireless Sensor Networks (WSNs):* WSNs consist of small nodes with sensing, computation, and wireless communications capabilities, and of special entities named sink, responsible for collecting and storing of the sensed data. The enormous interest in WSNs is fundamentally due to their reliability, accuracy, flexibility, cost effectiveness, and ease of deployment characteristics. Typical uses of sensor networks are environmental monitoring, event surveillance, or target detection. Hence, the majority of the time, the communication is one-to-all (i.e., sink to sensors) or many-to-one (i.e., sensors to sink). These networks are typically considered homogeneous as a single deployment typically consists of the same type of sensor devices. Nevertheless, some networks with medium heterogeneity constitute exceptions, where more capable and potentially static or mobile devices are used as actuators or sinks. It is generally assumed that WSNs are large-scale (e.g., hundreds or thousands of nodes) and that sensor nodes are strictly resource-constrained (e.g., in terms of energy, computation capacity, or memory). Thus, mechanisms for optimizing energy consumption and for the delivery of sensed data constitute important requirements and may affect node cooperation in the network. For instance, as nodes drain their batteries, they might refrain from participating in certain network functions, such as routing. Or still, depending on the mobility features of the sink, nodes might cooperate in order to guarantee that the data would be efficiently retrieved by the sink.
- *Disruption Tolerant Networks (DTNs):* In such networks, high degree of nodes mobility and unreliable links raise different communication challenges since contemporaneous path may never exist between a sender and a receiver and/or device reachability may be highly variable. Examples of DTNs include disaster response, underwater sensor, vehicular networks, and pocket-switched networks (in conferences, urban areas, or campus), which provide connectivity to users that carry their portable devices from one connectivity island to another. In particular, it was shown that humans have pattern of connection between themselves in the network that are neither purely regular nor purely random, describing a complex dynamic network displaying substantial non-trivial topological features. Due to nodes mobility, store-and-forward techniques can ensure eventual communication between any two nodes. In particular, in a DTN, the transfer of messages custody needs to be provided by nodes. Until a forwarder opportunity arises, a node may need to store multiple messages in its buffer. It is also possible that only one contact is available at a time and it has not enough resource to receive all custodies. In these typical scenarios of DTNs, some of the challenges to be considered by new DTN forwarding protocols are the following. First, due to limited duration of each contact, it is important to determine which and in what order the messages should be forwarded when an opportunity arises. Second, if more than one contact is available at any given time, the most promising contact(s) to where each message should be forwarded to, has to be determined. If, we consider infinite buffer and network bandwidth, the greater is the number of replicated messages, the better is the chance of that message is delivered to its destination. Nevertheless, resources are usually scarce in DTNs, making it necessary to determine in a dynamic way, the number of messages' copies that should be forwarded to contacts. Finally, if a buffer achieves its storage capacity and a new message has to be received, it is important to correctly determine which message should be dropped to accommodate the new one, while limiting the impact on the reliability of the dropped messages [13]. In DTNs, any type of communication requiring data delivery is a complex task – i.e., routing, data collection or dissemination – and should leverage communication opportunities in temporary connection.
- *Cognitive Radio Networks (CRNs):* Such networks are composed of fully programmable cognitive wireless devices (i.e., secondary cognitive users –SUs). The motivation behind CRNs comes from the fact that the unlicensed portion of the spectrum becomes increasingly overloaded due to the growing number of wireless devices and mobile users. Recent studies, however, have shown that while a small portion of the frequency spectrum is overloaded, a large part of the frequency spectrum licensed to primary users is being under-utilized or never used at all. Cognitive radios allow thus, the reuse of under-utilized portions of the frequency spectrum by secondary users (SUs) in a non interfering manner with primary users (PUs). To achieve this capability a SUs should be able to identify white holes, i.e. non-utilized frequency channels in a specific timeslot that are available for communication. Once identified the white holes, SUs have to dynamically adapt their channel access method, spectrum use, and networking protocols as needed for suitable network and application

performance [14]. In multi-hop ad hoc cognitive radio networks, where channel access should be performed in a completely distributed way by the SUs, cooperation between SUs is hard to achieve and data delivery support constitutes a challenge networking task. Hence, emerging needs appear in terms of utilizing connectivity opportunities and adapting communication to the constant changes in the networks.

1.2.2 WSONs' peculiarities and challenges

In all types of WSONs, the main network dynamics emerge from a variety of sources, such as mobility (that also affects link reliability), limited and variable capacity of wireless links, resource constraints, resource failures, and the spontaneous nature of the topology. Some additional peculiarities of WSONs include the possible lack of geographical positioning infrastructure, reliance on node cooperation, and high heterogeneity in resources or in traffic types, as detailed hereafter:

- *Distributed nature:* The lack of a centralized management requires autonomous operation of nodes. This asks for local coped decisions through simple neighborhood consensus, in order to limit communication overhead. Hence, WSONs solutions should rely on node cooperation, where information and management responsibilities should be completely distributed among the nodes in the network;
- *Varying and unreliable links:* Variables such as obstructions, interference, environmental factors, and mobility make determining connectivity a priori difficult in WSONs. Thus, the low link reliability and its possible asymmetry require that solutions designed for WSON be fault tolerant and adaptable to connectivity changes;
- *High heterogeneity:* In WSONs, it is likely that nodes are heterogeneous in their resource characteristics (such as memory availability, computation capacity, and transmitting power) and type of generated traffic. This requires solutions considering different bandwidth constraints and varying operating conditions of nodes (e.g., in terms of resource limitations);
- *Constrained resources:* WSONs participants often have limitations in memory, processing, and above all power. In this case, optimized solutions are strongly required to limit energy consumption and communication overhead. This also asks for (1) local coped decisions and (2) the distribution of information and management responsibilities among the nodes in the network;
- *Security vulnerability:* Due to their infrastructure-free and non-authority capabilities, WSONs are inherently insecure. In addition, transmissions are generally in broadcast mode, which opens the possibility for unauthorized nodes to listen to the traffic. Malicious nodes can then perform attacks in a much easier way.
- *Flexibility in route selection and dynamic-network management:* Node dynamics imposes changes on the topology, which in turn, implies more complex management algorithms for topology maintenance and routing. For instance, the dynamic nature causes routes to be unstable and make routing a resource-greedy operation: The more distant the corresponding node, the more expensive the communication. In this way, flexibility in route selection and simpler dynamic-network management are required;
- *Communication paths determined by physical location:* Due to the wireless link properties of WSONs, neighborhood is imposed by the physical location of nodes. Solutions should then rely on nodes' physical immediate neighbors for communication. Additionally, nodes mobility may require the use of opportunistic communication, where the monitoring of nodes' encounters and/or changing network conditions (e.g., link quality) is required for inferring resource management and/or forwarding decisions.
- *Data-centric abstraction:* Following the application requirements, some WSONs (e.g., WSNs or DTNs) have stepped away from the Internet's point-to-point address-centric communication model and have adopted data-centric abstractions. In the data-centric paradigm, the nature of the data is more important than the identity of the source. In particular, routing decisions are taken primarily based on the type of data. This model of communication clearly favors in-network data processing and aggregation, in addition to imposing changes in the way routing and storage are performed in the network.

This is by no means an exhaustive list. These WSONs peculiarities have a direct impact on robustness, reliability, performance, and lifetime of the services to be offered in such networks. For instance, mobility implies a dynamic topology, which, in turn, implies more complexity for location management, topology maintenance, and routing. Therefore, *WSONs require adaptive services that can handle such changes*, which is the topic of next section.

1.2.3 Adaptive services

Adaptivity provides high flexibility under different operating conditions but, poses significant challenges in terms of the design of such services. We define here a *space* of adaptation services based on different network dynamics. We categorize these services as *core* and *network-level* services:

- *Core services.* They are defined here as services that deal with network dynamics at the node level, where nodes govern their operation based on changes. We classify *location management, neighborhood discovery, and resource management* as core services, as they allow nodes to monitor and adapt to changing local conditions, and thus, to continue their participation to the network. Essentially, we consider the location, the neighborhood, and the available resources of a node, as the main factors that affect the node communication capability in the network. Clearly, a static network would not require location updates to maintain network connectivity. Similarly, if all nodes are tethered, have high CPU power, high bandwidth and no memory constraints, then there is also no need for resource management. In contrast, scenario context and external conditions may affect nodes and their wireless links and hence, call for adaptive services. For instance, mobile users in a campus environment (i.e., context) may require a location management or neighborhood discovery solution to facilitate communication. In this way, several network-level services can be built upon these core services, where routing, topology management, and data dissemination are the most straightforward examples.
- *Network-level services.* They are defined here as services that require collaboration from nodes to perform a specific task for the network and thus, adapt to changes on how nodes relate to each other. The network-level services typically rely on the core services. We categorized network-level services as *topology management, routing and forwarding, and data management (e.g., data dissemination and data collection).*

Additional categorizations are also possible. Services can be categorized based on whether they need *implicit* or *explicit* feedback [15]. Services that work with *implicit feedback* require a built-in functionality of passive monitoring and deriving information about the current conditions. For instance, a node can monitor the data traffic of its neighbors and based on the received signal strength in the monitored period, make decisions about whether its neighbors are moving or not. On the other hand, services that rely on *explicit feedback* require external support for providing such information in a pull/push-based manner. For instance, to monitor its neighborhood, a node might collect periodic hello messages from its neighbors and depending on these messages make decisions about neighbors' presence. Based on our example, a node relying only on *passive* monitoring might conclude wrongly that a neighbor has moved away, while the node is still in the given neighborhood but currently not sending any data. Hence, *implicit feedback* can only provide local and less accurate information through passive monitoring, which further might be difficult to interpret. On the other hand, although *explicit feedback* provides more accurate information, it has the disadvantage of higher overhead as well as requiring some sort of external support (e.g., nodes should know they should be sending periodic hello messages).

Furthermore, we can categorize services onto ones that adapt *periodically* and ones that adapt *on-demand* [15]. A *periodic adaptive* service regularly assesses the operating conditions and adapts if needed. In contrast, an *on-demand adaptive* service may trigger some changes when a performance parameter drops below a value or when more resources are discovered, characterizing an event-driven adaptivity to varying conditions.

Figure 1.1 depicts our categorization for adaptive services.

1.3 Contributions on adaptive services

As can be noted with the previous descriptions, WSONs have intrinsic characteristics and consequently, require particular solutions that set them apart from traditional "graph-based" networks. More specifically, the basic shift of paradigm between traditional networks and WSONs is related to the communication

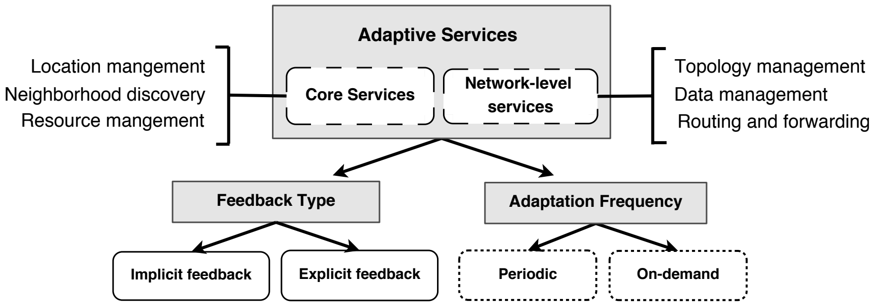


Figure 1.1: Categorization of adaptive services.

characteristics. Besides, each WSON has its own particularities, requiring adaptive solutions oriented to their characteristics, involved deployment scenario, and application.

Influenced by such observations, my research interests after my PhD focused on the different types of wireless self-organizing clouds expected to appear at the edges of the Next Generation Internet and tuned to their new requirements. The research activities I have been performing involve efforts to provide both *adaptive core and network-level services for WSONs.* Hereafter, I summarize such efforts, while detailing the provided types of services in each defined category. Fig. 1.2 also shows the classification of my research contributions (and of my colleagues) according to these categorized services.

1.3.1 Core services

We classify core services in *location management*, *neighborhood discovery*, and *resource management*, which are briefly described hereafter. For a more detailed survey, please refer to our work [12].

- *Location management:* In WSONs, network dynamics is the key characteristic that affects the location of nodes. Therefore, location management is the key to adapt the network behavior in face of changing conditions. We classify the adaptive location management approaches as: (1) Dissemination-based approaches, which distribute the location information throughout the network in an unstructured way [16, 17], and (2) Rendezvous-based approaches, which rely on specific rendezvous points to publish the location information of each node in the network [18, 9].
- *Neighborhood discovery:* The execution of most network operations demanding nodes coordination (e.g., routing or topology control) in WSONs requires knowledge of neighbors. In WSONs, network topology may change over time, requiring nodes to continuously discover or keep track of other nodes within their communication range. When considering *single channel neighbor discovery*, the literature offers approaches that are probabilistic [19] or deterministic [20] and that are based on active discovery methods (i.e., request feedback of neighbors) or have to control the transmission of discovery messages. In the multi-channel category, most neighborhood discovery strategies are passive, i.e., only schedule the listening periods of nodes, and take benefit of the periodic beacon transmissions of the MAC [21, 22, 23].
- *Resource management:* In the literature, the majority of resource adaptive approaches focus on one particular resource, among which energy is the most commonly studied. Energy management in WSONs is mainly focused on the energy consumption of the wireless network interface as it is one of the main consumers. We categorize the current adaptive approaches to energy management as (1) Adaptive communication approaches, which rely on transmission power control to reduce the transmission power level and on rate control approaches to save the time spent in communication [24], and (2) Adaptive sleeping approaches, which rely on finding a duty cycle that fits the current network traffic [25].

Concerning core services, my performed research activities relate to the provision of: location [2, 8, 9] and neighborhood discovery services [21] (cf. Fig. 1.2). In terms of location service, the provided solutions concern the ones designed during my PhD, which rely on explicit feedback and perform on-demand adaptation. Neighborhood discovery services were considered during the time I spent working as an invited researcher at TU-Berlin (from Nov. 2009 to Oct. 2010) in Germany. In our neighborhood discovery

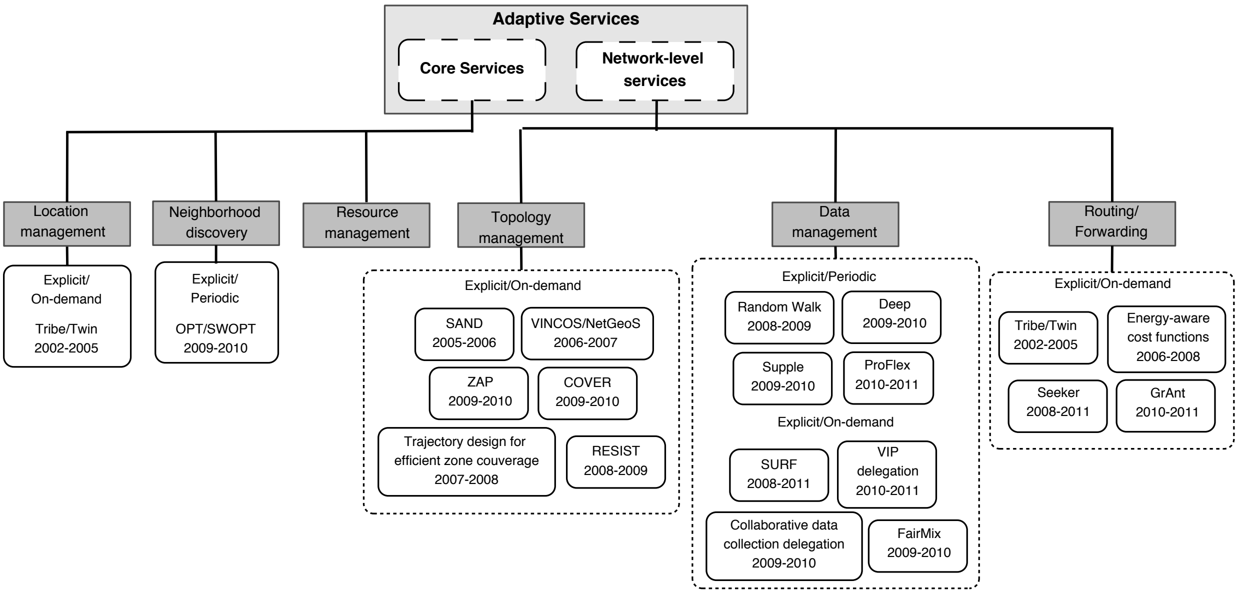


Figure 1.2: Classification of research activities according to the category of provided services.

work, we focused on scenarios involving wireless personal area networks (WPAN), such as: body sensor networks requiring health- and wellness-related patient monitoring or situations requiring opportunistic message propagation. We have investigated optimized discovery of IEEE 802.15.4 static and mobile WPANs operating in multiple frequency bands and with asymmetric beacon intervals. Our solutions, named *Opt* and *Swopt*, deal with the asynchronous and multi-channel fast discovery problem [21]. Such solutions rely on explicit feedback and perform periodical adaptation.

1.3.2 Network services

Due to page limitation, the remaining of this document is, however, devoted to the research I have been conducting around adaptive network-level services, which I roughly classified as topology management, data management, and routing and forwarding. Fig. 1.2 locates my research contributions (and of my colleagues) according to this classification. Such works were performed during the time I have been working at INRIA, as a Post-Doc (2005-2006, INRIA Rennes) and later as a Research Scientist (2006-).

The network-level services are briefly described hereafter, but a more detailed description can be found in [12].

- *Topology management:* Topology management can be defined as shaping the network topology to achieve a certain objective, such as better connectivity, fault-tolerance, or energy management. The network topology can be modified (1) through node adaptation (e.g., by imposing a hierarchy in the network through clustering [26] or removing nodes from the network graph by powering them off [27]); (2) through link adaptation (e.g., using transmit power control [28], smart directional antennas [29], or intelligent channel assignment [14]); and (3) controlled mobility (e.g., in actuator sensor networks), which affects both the presence of nodes and links, as well as the quality of links in the network graph [30, 31].
- *Data management:* Data is an important element in any computer network. We classify data management approaches according to the following tasks: (1) *data collection*, where data flows originated in the network are directed towards an external entity responsible for collecting the data (e.g., gathering of monitored data) [32, 33]; (2) *data dissemination*, where data flows are directed towards the network (e.g., node configuration, distributed data storage) [34, 35, 36, 37]; or (3) *data processing*, where data flows are manipulated within the network (e.g. data compression) [38] in order to prepare them for data collection or dissemination.
- *Routing and forwarding:* Any routing protocol for WSONs can be defined as a composition of three fundamental architectural blocks: identification, location, and forwarding [39]. The ultimate goal

is then the effective transmission of data, implemented by the forwarding scheme, which dictates the quality of the selected route. We classify the routing approaches in the literature according to the factors that drive their adaptation: (1) Mobility, which groups proactive, reactive, and hybrid approaches intended for connected networks [40]; (2) Connectivity opportunity in scenarios with intermittent connectivity, which meets the challenges of DTNs and groups controlled flooding, utility-based, and social-behavior-based approaches [41, 42] and (3) Energy consumption, which groups approaches using energy management as a core service [43].

In particular, my main concern in the context of network-level services has been the provision of *adaptive networking support for successful data exchange/delivery in the WSONs*. According to the WSON application type, successful data delivery might involve: network topology's shaping, collection (e.g., by mobile infrastructure nodes), forwarding, or dissemination. Such data delivery activities might also relate to different way of communication (such as n to 1 – as in WSNs –, 1 to 1, or 1 to n) and interaction patterns (such as static-to-static, static-to-mobile, mobile-to-static, or mobile-to-mobile interactions) between network nodes.

To count such communication diversities and the problem of successful data delivery in WSONs, I have been considering *flat* and *two-tier* networking support solutions. The first case only requires cooperation between nodes in the network. Instead, the two-tier support consists of enabling placement, refinement, and/or repositioning of infrastructure nodes as the network evolves depending on the (i) mobility patterns of nodes, (ii) traffic characteristics, as well as (iii) conditions of the infrastructure nodes (e.g., remaining battery lifetime, buffer capacity).

The Table 1.1 places my research work (and of my colleagues) according to the different types of interaction patterns and networking support type of WSONs. It is worth mentioning that adaptive support for successful data delivery has been the main concern of all these works. The specific topics I chose to discuss in this manuscript are in bold.

Table 1.1: Positioning of my contibutions considering WSONs networking support.

Interaction Patterns	*Networking support type* Flat	Two-Tier
static-to-static	**VINCOS/NetGeoS** [26, 44, 45], **SAND** [27], Energy-aware cost functions [47, 48], **FairMix** [35, 49, 34], RESIST [50]	ProFlex [46]
static-to-mobile	Random Walk-based approach [51], **DEEP** [32], **Supple** [33], **Trajectory design for efficient zone coverage** [31]	–
mobile-to-mobile	**Seeker** [52, 41], ZAP [14, 53], **GrAnt** [55], SURF [56]	**Cover** [30], **VIP delegation** [36, 54], Collaborative data collection [37]

1.4 Outline of the manuscript

The remainder of the manuscript is composed of four chapters. Chapters 2, 3, and 4 present my contributions according to the three groups of previous mentioned network-level services, i.e., topology management, data management, routing and forwarding. Due to page limitation, I do not get into their details. Instead, in each chapter, I present the motivations and main ideas of my contributions and try to give a global picture of the studied problem and involved challenges. In order to concentrate on the characteristic's description of the proposed protocols, I deliberately choose not to present any performance result. Nevertheless, all the presented protocols were validated under simulation and/or under real experimentation (in the case of the neighborhood discovery service described in [21]). For more details on each of the proposals, the reader is invited to refer to the respective publications. In Chapter 5, I summarize the presented research activities and list a number of perspectives for future work.

It is worth to mention that some of other contributions that are related to the research presented in this manuscript could not be described. Nevertheless, a complete view of my research activities can be found on my personal webpage at `https://sites.google.com/site/alinecviana/`.

2 Adaptive topology management services

The research contributions presented in this chapter concern topology management services performed through node adaptation and controlled mobility.

Although not detailed here, we have also conducted work on topology management through link adaptation, where a channel assignment strategy is proposed for cognitive radio networks. In our proposed ZAP protocol (in collaboration with P. R. Walenga Junior, M. Fonseca, A. Munaretto, and A. Ziviani), a simple heuristic for channel distribution among cognitive radio users is performed based only on local (neighborhood) knowledge, with the goal of mitigating the interferences among simultaneous transmissions. More details concerning its design and evaluation are available in [14, 53].

2.1 Topology management through node adaptation

The topology of a network can be most easily modified by imposing a hierarchy over the network through clustering or by powering nodes off. In general, such modification process allows better address assignment, bandwidth utilization, and energy consumption, as most communication remains local. In this section, the protocol *SAND* (Self-Organizing Active Node Density) [27] and the systems *VINCOS* (VIrtual Networked COordinate System) and *NetGeoS* (Networked Geometric Structuring approach) [26, 44, 45, 57, 58] are presented, which are related to energy-conserving topology management and geometric self-structuring of nodes in wireless sensor networks (WSNs). They all provide flat networking support solutions (where the adaptation required in the topology management process is performed by the sensor nodes) and rely on with explicit feedback on-demand adaptive service.

The considered context is an area monitoring application, which is one of the most typical applications of WSNs. Area monitoring consists in deploying a large number of sensors in a given geographic area, for collecting data or monitoring events. It is not unusual that in this situation, human intervention is not feasible. Sensors are then thrown in mass, for example from a plane, and must be able to form a network and to operate in a decentralized self-organized manner, maintaining connectivity and area monitoring as long as possible.

2.1.1 Energy-conserving topology management

(Publication [27]. In collaboration with E. Le Merrer, V. Gramoli, A.-M. Kermarrec, and M. Bertier)

Mechanisms for energy optimization in WSNs constitute an important requirement. The sensor network lifetime is defined as the period during which the routing fidelity and the sensing fidelity of the network are guaranteed. Guaranteeing sensing fidelity means that any monitored stimulus in the area will always be sensed by at least one sensor. Routing fidelity means the existence of a path between any sensor node and at least one base station. Our goal in this work was then *to leverage node redundancy in WSNs to reduce and distribute the computational and communication energy consumption of the network between sensors*. To this end, we proposed a simple and adaptive energy-conserving topology management scheme, called *SAND (Self-Organizing Active Node Density)*. SAND significantly extends network lifetime by reducing nodes activity. It is fully decentralized and relies on a distributed probing approach and on the redundancy resolution of sensors for energy optimizations, while preserving the data forwarding and sensing capabilities of the network.

Some topology management techniques in the literature [59, 60], have similar goals to those of SAND: they trade network density for energy savings while preserving the forwarding capacity of the network. Nevertheless, they do not exploit the absence of traffic in the active sensing state. On the other hand, some approaches have been proposed for exploiting the area coverage problem in sensor networks [61]. These solutions assume, however, that the sensors are aware of their own positions; generate high communication overhead; or require nodes to memorize the positions and the decisions of their neighbors in order to make appropriate monitoring decisions. In SAND, however, nodes need a small amount of information (e.g., only a partial neighborhood discovery) and perform a low processing overhead to take their activity decisions, while no positioning is required.

We considered that during network lifetime, sensor nodes can alternate their energy consumption between four states: (1) sleep, where all hardware components are powered off, (2) sensor-only, where only sensor and some pre-processing circuity are powered on, and (3) gateway and router-sensor, where all hardware components are powered on. No node location information is required. We explore density determination by assuming that nodes communicate only by 1-hop broadcast toward nodes in their neighborhood, corresponding to their transmission range. SAND performs then the energy-aware topology management by controlling the routing and the sensing fidelity during the network lifetime. Such fidelity control is described in the following.

Routing fidelity: The forwarding nodes distribution is performed in two consecutive phases. The first one distributes nodes in router-sensor state uniformly in the network. The second one consists in connecting close router-sensors by selecting nodes to switch to the gateway state.

In particular, in SAND, the router-sensor nodes periodically send `Hello` messages containing their current state and identifier (as a tie breaker), and a timestamp t_s, at each $\Delta > 2\delta$, where δ is an upper bound on the communication delay between any two nodes in the same neighborhood. Observe that we do not focus on specifying the message retransmission in case of collision; we rather assume that this is implemented at a lower layer. The timestamp t_s of node i informs how long i has been in the current state. Each sensor-only node in the network checks if there is a router-sensor node in its immediate neighborhood, by listening during a timeout T_{on} with $T_{on} > \Delta + \delta$. If no router-sensor node is detected, then the sensor-only node becomes a router-sensor node. If a router-sensor node detects the presence of another router-sensor node in its transmission range with a higher timestamp than its own, it comes back to the sensor-only state.

As a router-sensor has no other router-sensor in its area so the first phase guarantee only a good distribution of router-sensor, the second phase elects gateways to connect them. A sensor-only node becomes gateway if it detects two routers and no gateway with a lower timestamp than its own and that already makes a link path between those two routers. A gateway node informs with a period Δ about the routers it sees. If a gateway detects an other gateway which connects the same router-sensor and with a higher timestamp than its own, it comes back to the sensor-only state.

In summary, SAND presents two interesting properties borrowed from graph theory. First, to help routing to a sink node, each node is in the neighborhood of a router-sensor node or is itself a router-sensor node. Second, to prevent energy consumption waste, a subset of nodes becomes router-sensor nodes. There are solutions for related problems known as vertex cover and minimal dominating set, which guarantees activated sensors to form such a set. The minimality problem of the aforementioned solutions might involve many state changes each time a router-sensor crashes. Here, we rather ensure that the sensor-router nodes satisfy both *domination and independence properties.* This helps at reducing, yet making it sufficient, the number of router-sensor nodes, while this number has not to be minimal. Roughly speaking, the router-sensor nodes satisfy (i) *dominance:* all nodes are either router-sensor or a neighbor of a router-sensor and (ii) *independence:* no router-sensor node is a neighbor of another router-sensor node. For a proof on the convergence of the algorithm to a configuration verifying both properties under system stabilization (i.e., dynamic events stop), we invite the reader to refer to [27].

Sensing fidelity: SAND allows the control of the sensor-only nodes resolution in each target area, while performing the sensing load distribution among nodes. To this end, router-sensor nodes are in charge of selecting nodes to switch to sleep or sensor-only state. This selection depends on the envisaged reliability degree of the monitored area. Thus, nodes that are for a long time in the sensor-only state will be selected to switch to sleep state, and vice versa. Each sensor-only node periodically turns on its radio and sends `Hello` messages containing its current state and its estimated lifetime (el). Nodes in sleep state also periodically turn on their radio, but never send messages.[1]

Upon reception of el of sensor-only neighbors, router-sensor nodes compute the average (by considering the el of the last switched-to-sleep nodes too) and the standard deviation (by considering only the sensor-only nodes that have lower el than the resulting average). Finally, router-sensor nodes send 1-hop "`switch-to-sleep`" order messages to sensor-only nodes that have their el level lower than the resulting standard deviation. The switch of sensor-only to sleep state is performed as soon as another sensor-only

[1] We assume their estimated lifetime is the same of the last el sent when they were in the sensor-only state.

node appears in the monitored area. Sensor-only nodes also turn on their radio if any local collected data has to be transmitted to the sink (e.g., in case of full memory resource).

Router-sensor nodes also control the switch from sleep to sensor-only states, by sending "`switch-to-sensor`" order messages. In this case, nodes in the sleep state switch to the sensor-only state if they have (1) their radio turned on and (2) their *el* level is higher than the computed average. In addition, like sensor-only nodes, nodes in the sleep state with the radio turned on, verify their router-sensor node connectivity. If no `Hello` message from a router-sensor node is received, a sleep node becomes a sensor-only node and then, based on the SAND bootstrap procedures, can switch to the router-sensor or gateway state.

Finally, we claim that a sufficient density of sensor nodes will provide enough gateway candidates to ensure the connection between two close router-sensors. SAND does not determine the optimal minimum number of forwarding nodes to maintain sink connectivity, ensuring then, that there are several paths between any node and at least one sink. This redundancy makes the routing fidelity more resilient to failures. We have conducted a number of simulation experiments using a discrete time-based engine. Our experiments show that SAND guarantees for a longer time, (1) the existence of paths between any sensor node and at least one sink node in the network and (2) the correct sensing of stimulus in a monitored sensor network. SAND improves considerably network lifetime proportionally to node density, at the price of the slightly increasing paths length from sensor to sink nodes.

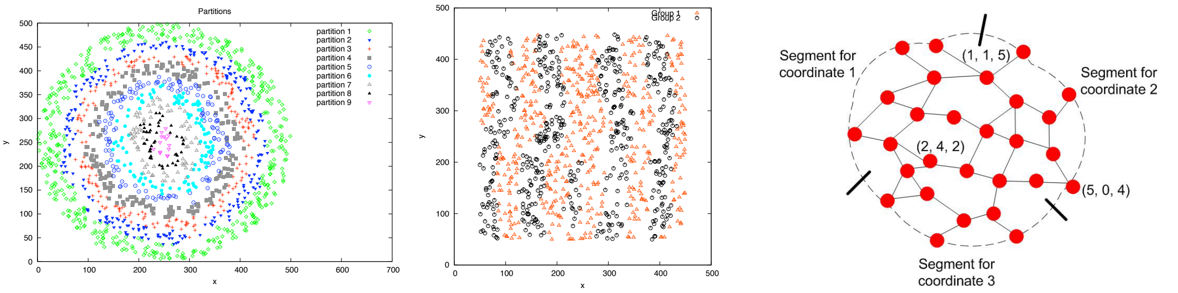
2.1.2 Geometric self-structuring topology management

(Publications: [26, 44, 45, 57, 58]. In collaboration with G. Trédan, A.-M. Kermarrec, A. Mostéfaoui, M. Raynal)

Self-structuring refers to the ability of a set of distributed nodes to allow a specific structure to emerge from scratch, while requiring as few as possible specific initial information or service (such as the number of nodes, their geographical position, the existence of a global positioning system, the existence of local landmarks, etc.). Self-structuring is an important dimension of a system autonomy (especially in terms of scalability issues), where common operations such as forwarding, load balancing, leader election, or energy consumption management are required [62]. An example of self-structuring in this kind of networks includes the partitioning of an area in several zones for monitoring or organization purposes, as shown in Fig. 2.1(a) and 2.1(b).

The complexity of a self-structuring mechanism strongly depends on the amount of knowledge that is initially provided to nodes in the network. If any node in the system has a complete knowledge of the system, structuring the network is trivial. Otherwise, if nodes are only aware of their own neighborhood, ensuring that a given structure emerges from individual decisions is challenging. In short, the autonomy degree of a networked system is inversely proportional to the external knowledge required to structure the network. It is, however, crucial to come up with a reasonable trade-off between autonomy and overhead. To counter these issues, we presented a robust structuring mechanism, named *Networked Geometric Structuring* (NetGeoS). NetGeoS leads to geometric organization that can be deployed from scratch in a networked system where the initial knowledge of each node is limited to its own identity and communication range. Nodes can then be assigned to different adaptive behaviors based on the established organization. To the best of our knowledge, this was the first geometric structuring autonomous system deployed upon those conditions in the literature.

The organization laws dictated by NetGeoS are motivated by the "divide and conquer" paradigm, allowing nodes to get different adaptive behaviors based on the established organization. Nevertheless, network structuring becomes a very challenging goal, as soon as neither positioning referential, boundary delimitation, nor density distribution is provided. One solution for this is to allow the nodes to access a virtual coordinate system from which they can obtain a coordinate assignment, on top of which adaptive behaviors can be designed. Virtual coordinates better reflect the real network connectivity, and can consequently provide more robustness in the presence of obstacles. In addition, since virtual coordinates are relative to physically close nodes' neighborhood, they can tolerate more environment dynamism than absolute geographic coordinates. Despite having clearly defined outlines and presenting good approximation solutions, previous work on virtual coordinates [63, 64, 65, 17] are computationally- (and message) costly, hardly practical in wireless sensor networks, or still require the nomination of well placed entities



(a) (b) (c)

Figure 2.1: (a)-(b) Self-structuring example given by NetGoS and (c) A 3-dimensional ($d = 3$) virtual coordinate system given by VINCOS system: $(2, 4, 2)$ means that the corresponding node is at distance 2 of both the borders 1 and 3, and at distance 4 of the border 2.

in the systems to work as anchors or bootstrap beacon nodes. Instead, we proposed the VINCOS system for WSNs (described in the following), which does not rely on any anchors, position-aware landmarks, or signal measurement.

From anarchy to virtual coordinates: A coordinate system provides each node with a "position" that is both individual and globally consistent. In VINCOS, each node is initially configured with the parameter d indicating the dimension of the coordinate system (number of coordinates). The virtual coordinates of a node i are consequently a tuple $(x_1, \ldots, x_d)$, where x_j is the projection of i on the jth axis of the d-dimensional virtual space. The virtual d-dimensional space is defined as follows. The border of the geographical area covered by the nodes is partitioned into d *"segments"*. This *border segments* can have the same size or different sizes (we assume in the work that they have the same size). This depends only on the algorithm and may easily be changed. Let us consider any axis j, $1 \leq j \leq d$, of the coordinate system. The coordinate x_j is the length, in hops, of a shortest path from the node i to the border segment j (i.e., to the closest node on that border segment). This is illustrated on Fig. 2.1(c) through a simple example, where the virtual coordinates of three nodes are indicated.

As previously mentioned, the coordinate assignment of VINCOS depends on an accurate definition of the border segments (i.e., the d segments dividing the border of the considered geographical area). To define the border segments, VINCOS relies on a *belt* construction mechanism. The resulting belt, defined as a set of *border-belt nodes* (nodes located at the perimeter of the area), is a connected structure that (1) enables communication among border-belt nodes along two different paths; (2) allows an order assignment to these nodes; (3) is one-hop wide; and (4) has proportional size wrt the network size[2]. This ensures that, given the broadcast communication pattern, all message forwarded along the belt (complete round) reaches every border-belt node.

Nevertheless, discovering a connected border at a low cost and in an accurate way is a challenge task and requires the nodes being able to acquire some consistent knowledge of the network. VINCOS provides nodes with a fully decentralized way of acquiring that knowledge. The protocol is composed of four consecutive phases: (1) initiators detection, used to define the local "maximal" nodes from a density point of view; (2) border score definition, used to identify perimeter bootstrap nodes; (3) border-belt construction, used to identify border-belt nodes; (4) coordinates definition. The total communication cost imposed by VINCOS sums up the cost of each phase (recall N is the total number of nodes of the system, y and z are the numbers of initiators and perimeter bootstrap nodes, respectively, and d is the size of the coordinate system): $O((2 + y + d) * N + 4 * z\sqrt{N})$. Similarly to communication cost, the total expected convergence time is $2 + (2\sqrt{2} + 4)\sqrt{N}$ interactions.

From virtual coordinates to geometric structuring: Geometric structuring can be defined as a logical partitioning of the network into geographical zones for application-dependent purposes. Partitions are then used to fulfill specific applications requirements or systems properties, for example. Formally,

[2]A too small belt could generate inaccuracies in the coordinate system resulting in many non-neighbor nodes being assigned to the same coordinates.

let $\mathcal{K}$ be the system nodes coordinate space (in our application, $\mathcal{K} \in \mathbf{N}^d$), and let p be the number of partitions. Let c_i and p_i be the coordinates and the partition number of node i, respectively. Then a geometric structuring function is a function f s.t. $f : \mathcal{K} \rightarrow \{0, \ldots, p\}$, where $f(c_i) \rightarrow p_i$. Let us observe that such a definition allows any node to compute the partition number of any other node whose coordinates are known: *each node has a global foresight of the system layout.*

To illustrate our purpose, consider the *target-like partitioning*, as shown in Fig. 2.1(a). This is a straightforward structuring to achieve using $d = 1$. In this structure, each node gets as a partition number, its minimum hop distance to the border ($f : \mathbf{N} \rightarrow \mathbf{N}$): $f(x_1) \rightarrow x_1$. This can be a useful structure for tracking application for example, where all the inner rings could be set in sleep mode, the only active partition being the border. Whenever a node from a partition p senses something, it wakes up the partition $p + 1$, so that the network is gradually woken up. In another example, parallel vertical or horizontal lines can be used to select well distributed nodes in the network to be responsible for performing data aggregation or replication structure to be later visited by a mobile sink. For this, consider the vertical line partitioning shown in Fig. 2.1(b), where each node i belongs to the vertical line at each j_v jumps, resulted from ($f : \mathbf{N}^4 \rightarrow \{0, 1\}$): $f(x_1, x_2, x_3, x_4) \rightarrow max(x_1, x_3) \mod j_v$.

Full details on the algorithms related to the coordinate system and to the geometric structuring are available in [26, 45] along with a simulation evaluation using a discrete event simulator. Simulation results attest the accuracy of both the network coordinate system and the geometric structuring. The structuring is surprisingly well achieved, yet, in a fully decentralized way, and regardless of the shape, distribution of nodes and presence of voids in the network.

2.2 Topology management through mobility

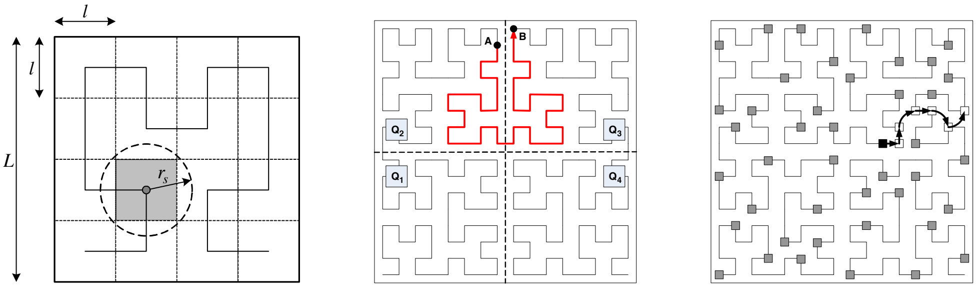
In networks using devices with controlled mobility (e.g., sensor actuator networks), the ability to move nodes creates a great potential for adaptivity as the network topology can be changed to meet deployment requirements (e.g., network coverage) as well as improve network performance (e.g. latency or frequency requirements) and fault-tolerance. In this section, the work related to the trajectory design for efficient zone coverage [31] and the protocol *Cover* [30] are presented. Both contributions focus on the deployment of solutions for zone coverage with mobile nodes, designed to periodically monitor a geographic area or to cover mobile sensor nodes (targets). While the first solution relates to flat network support approach, the latter relates to a two-tier deployment solution, where the adaptation required in the topology management process is performed by infrastructure nodes covering mobile sensor targets. Additionally, both works rely on on-demand adaptive service and explicit feedback.

2.2.1 Trajectory design for efficient zone coverage

(Publication [31]. In collaboration with Marcelo D. de Amorim)

In this work, we have considered applications of mobile wireless sensor networks that require periodic coverage. This means that readings should be performed following some predefined value $f_{\min}$ that denotes the minimum frequency at which the whole target area must be sensed. Examples are temperature and humidity control. In networks using devices with controlled mobility, the ability to move nodes creates a great potential for adaptivity as the network topology can be changed to meet deployment requirements as well as improve network performance and fault-tolerance. Other works in the literature have shown *why* mobility is useful to increase coverage and to reduce energy consumption in relaying traffic [66]. In this work, we considered mobile integrated sensor/actor network and focus, however, on *how* sensor/actor nodes should move, in order to guarantee the coverage of all targets in the network in a timely and efficient way. This is a difficult problem, as many constraints exist in order to deal with the latency and coverage issues. The failure in visiting some target points result in data loss, while the infrequent visit of points result in large delivery delays.

We proposed to address these requirements by using the mathematical concept of space-filling curves. Space-filling curves guarantee the filling of a d-dimensional space by traversing every defined point in the space once and only once, in some particular order. In particular, we explored the advantages of the Hilbert space-filling curve [67]. The property of locality preservation of the Hilbert curve [68] leads to



(a) (b) (c)

Figure 2.2: (a) Hilbert curve of order 2, where the grey area indicates the basic cell to be covered of order. (b) Hilbert curve of order 4. Points A and B mark the maximum distance between any point on the curve and the border. (c) The curve is of order 4 hosting 43 sensor-actor nodes. Straight arrows represent steps where nodes carry the reading, while curve arrows show when nodes opportunistically send the reading to one of its neighbors.

better contacts and consequently, to better data forwarding opportunities. As detailed in our paper [31], this is a fundamental property to reduce delivery delays. Hilbert space-filling curves have been used in a number of other domains [69, 9]. Nevertheless, to the best of our knowledge, this is the first work that takes advantage of this kind of curve to get cadenced target coverage.

The strict definition of the space-filling curve allows sensor-actors' trajectories to be known in advance. In order to keep our focus on the design properties of our approach, we considered a regular-shaped target area without holes/obstacles. Once the Hilbert space-filling curve was chosen, we worked on how the system should be designed to support the requirements of the application. For this, we have firstly dimensioned the curve and then derived the number of sensors that are necessary to cope with $f_{\min}$.

The first step computed the most adequate dimension of the Hilbert curve that respects our constraints. The first condition is that the curve must be dimensioned in such a way that, by visiting all points defined in the space, the entire target area is covered. To this end, we computed the order θ of the Hilbert curve (i.e., the number of required sub-squares or cells N that divide the area) required for assuring a sensing coverage of radius r_s in a square area of size L: $\theta = \left\lceil \log_2 \left(\frac{L\sqrt{2}}{2r_s} \right) \right\rceil$ (see Fig. 2.2(a)). The N cells of the target area are traversed once and only once by the trajectory provided by the Hilbert curve. For the reading frequency $f_{\min}$ to be respected, the maximum distance between two consecutive sensor-actors with maximum speed v must be $d = \frac{v}{f_{\min}}$ and the minimum number of mobile sensor-actor nodes to be deployed in the area must be $M = \lceil \frac{D}{d} \rceil = \left\lceil \frac{f_{\min} L(N-1)}{v\sqrt{N}} \right\rceil$, where D is the length of the curve and equals to $D = \frac{L(N-1)}{\sqrt{N}}$ (since the curve is composed of $N-1$ segments of length equal to the side of a cell $l = L/\sqrt{N}$).

Another important requirement is to deliver readings to base stations within some pre-defined maximum delay that corresponds to the validity period of the reading. The basic algorithm for a node to deliver a measure to the sink is to store the reading until the sensor-actor reaches the border of the sensed area: i.e., any point on the curve whose falls in the set $\mathbf{B} = \{(0, y), (x, 0), (2^\theta - 1, y), (x, 2^\theta - 1)\}$. We call this algorithm *carry-and-deliver.* Although simple, the carry-and-deliver algorithm has the inconvenience of resulting in too long delays in the worst-case, named $\hat{\delta}$ (i.e., the longest distance between any point on the curve and the border). In a Hilbert curve of order θ, this corresponds to the segment connecting coordinates $(\omega/2 - 1, \omega - 2)$ and $(\omega/2, \omega)$, where $\omega = 2^\theta$ and equals to $\hat{\delta}_{\{\theta\}} = \sum_{i=2}^{\theta} 2^{2i-3}, \forall \theta \geq 2$ (see segment between points A and B in Fig. 2.2(b)). Note this longest distance grows exponentially with the order of the curve.

To address this problem, we proposed the use of a *opportunistic delivery* coordination algorithm that takes advantage of encounters between sensor-actors to reduce the average delivery delay of readings. At each position on the curve, the node decides either to keep the message or send it to one of the neighbors. More specifically, a node decides to send a message to a neighbor *if and only if the distance from this neighbor to the border is strictly shorter than the node's own distance to the border.* Opportunistic decisions are made on-the-fly by each sensor-actor and are based on their local neighborhood observation (see Fig. 2.3(c)).

We have performed numerical analysis to evaluate the presented ideas. Through these analysis, we have shown that this approach (1) guarantees the coverage of the entire target area, (2) limits the number of required mobile sensors in the monitored region, and (3) bounds the delivery delay of readings. Although our work is directly applicable in the context of square-shaped target areas, it raises interesting issues that should be taken into account while generalizing the proposed concepts.

2.2.2 Adaptive deployment for pervasive data gathering

(Publications: [30]. In collaboration with T. Razafindralambo, N. Mitton, M. D. de Amorim, and K. Obraczka)

In this work, we have focused on scenarios where end nodes (or targets) being monitored/tracked are mobile (such as vehicles, animals, humans, etc) and little a priori information about their mobility is known. It is then critical to design efficient protocols to support pervasive "any time, any place" services in these networked environments prone to connectivity disruptions. This is a fairly complex zone coverage problem and constitutes the main focus of this work.

To meet the abovementioned challenges, we designed the *Cover* protocol, an adaptive strategy for infrastructure node placement and (trajectory) control. *Cover* assigns geographical zones to mobile infrastructure nodes. To this end, infrastructure nodes constantly check the number of targets they cover and adapt their trajectory when necessary.

Area coverage has been investigated in the literature for a while. However, existing solutions have been designed to cover static points/targets. There are mainly three categories of works that focus on coverage optimization in the literature: 1) random deployment of sensors: which consists in a huge number of sensors being randomly deployed with activity scheduling or power control techniques being used to reduce the network density [70]; 2) off-line computation of sensor placements: which is based on network performance, connectivity and area coverage [71]. The works presented in these papers give an overview of possible off-line node placements and their coverage performances; and 3) sensor repositioning scheme: which mainly focuses on the sensor (re)positioning or online placement [72]. Some similar approaches to *Cover* also take advantage of node mobility to enhance the network connectivity [73] but in these approaches, the mobility of target nodes is known and controlled. For the interested reader, a complete state of the art has been provided by Younis and Akkaya [74].

This work relies on two-tiered architecture consisting of data-producing nodes (targets, named TN) and infrastructure nodes (named IN) whose main function is to provide network connectivity support. The main issue in such a context is then *how to deploy and manage the mobility* of the infrastructure nodes in order to guarantee that all the target nodes are covered while respecting the application delay constraints and balancing load. This means to guarantee that every TN meets an IN in a regular way and that encounters cannot be spaced of more than some maximum delay.

Model and assumptions. We assume that the target area to be monitored is known and is a square of size $L \times L$. We divide this area into C cells of size $l = L/\sqrt{C}$. In this area, we deploy N mobile TNs nodes and M INs. We denote v_{IN} (resp. v_{TN}) the speed of INs (resp. TNs). The cell C defines the minimum area that a static IN can continuously monitor. We consider IN has to monitor multiple cells (i.e., $M < C$): Let $Z(m_i)$ be the number of cells in the zone covered by IN m_i and $T(m_i)$ be the number of TNs in these cells (i.e., the number of TNs monitored by m_i). We also assume that there is a maximum number of TNs $T_{\max}$ that a single cell can accommodate, which limits the number of TNs covered by m_i to $T_{\max} \times Z(m_i)$. Two INs are considered as neighbors if the zones they cover share at least one border of size l. Fig. 2.3(a) illustrates one possible configuration for seven INs. In this example, m_2 has four neighbors (m_1, m_3, m_4 and m_6), while m_7 has only two neighbors (m_1 and m_6).

By definition, TNs report to INs. An IN can retrieve data from a TN if and only if they occupy the same cell at the same time. An IN computes its trajectory as a function of the set of cells it has to monitor. INs have to regularly visit $Z(m_i)$ cells – we call the fact of visiting all cells of a zone a *cycle* and define it as the *coverage* of the IN. We refer the frequency at which a IN visits a cell as $f_{\min}$. Finally, we assume that the time an IN spends in each cell is enough to retrieve data from at most a maximum number of TNs $T_{\max}$.

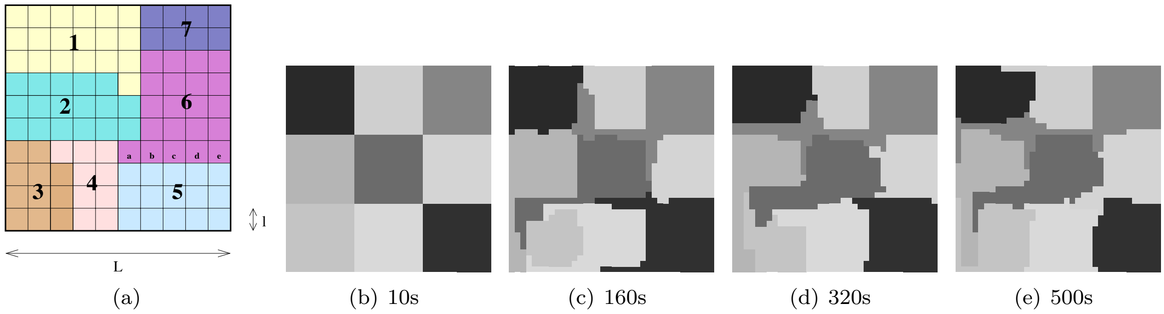


Figure 2.3: (a) Illustration of an area of 10×10 cells covered by 7 INs. The area covered by each of them is denoted by the node's identifier. (b)-(d) Example of zone evolution in time, with 9 INs and 250 TNs following a random walk mobility model.

Whole coverage. At bootstrap, we assume that INs are uniformly distributed (cf. [75]). In this way, the zone to cover, composed of C cells, is equally shared among INs. Since an IN can cover at most $Z_{\max}$ cells, the system needs a minimum number of INs, $M_{\min}$, such that $M_{\min} = \left\lceil \frac{C}{Z_{\max}} \right\rceil$. the number of cells an IN can monitor (i.e., $Z(m_i) \leq Z_{\max}$) has to be bounded with regards to the IN speed and the required reading frequency $f_{\min}$: i.e., $Z_{\max} = \frac{v_{IN}}{f_{\min}} \times \frac{1}{2l}$. Indeed, $\frac{v_{IN}}{2f_{\min}}$ represents the maximum number of cells that an IN can cover during a cycle of duration $\frac{1}{2f_{\min}}$. Given so, INs are assigned zones in a way that: (1) $\left\lceil \frac{C}{M} \right\rceil$ cells are assigned to $(C \bmod M)$ INs; (2) $\left\lfloor \frac{C}{M} \right\rfloor$ cells are assigned to $(M - (C \bmod M))$ INs. Indeed, at bootstrap, every cell is covered by exactly one IN.

TN balancing. The goal of *Cover* is to balance the number of TNs per IN in order to balance load. Each of the cells contains a certain amount of TNs. The number of TNs covered by IN m_i has to tend to an optimal value $T_{\text{opt}} = N/M$ in order to balance the load among INs. Indeed, if every IN covers T_{opt} distinct TNs, by definition, the whole set of TNs is covered and coverage sets are non-overlapping. The *Cover* algorithm runs in a distributed way at each IN and balances the number of TNs covered by each IN while ensuring the reading frequency and the coverage of the whole area. If the number of covered TN by a IN is $T(m_i) > T_{\text{opt}} + \theta$, this means that such IN monitors too many TNs. *Cover* than ensures that a cell will be delegated to one of the neighbors of this overloaded IN. Cells are naturally delegated one at once, till getting $T(m_i) < T_{\text{opt}} + \theta$ or having no neighbor IN able to receive an extra cell (refer to [30]). When assigning a cell to a neighbor, an IN gives preference to cells that allow keeping its zone connected and as compact as possible. By compact, we mean that the IN tries to minimize the size of its border. Since at each step of the algorithm an IN delegates a cell if and only if another IN accepts to monitor it, whole coverage is always guaranteed.

In [30], we showed through theoretical analysis and extensive simulations that (1) *Cover* converges when the target nodes are fixed and (2) that the number of target nodes per infrastructure node is close to the optimum at any time independently of the mobility pattern of the target nodes. The simulations were performed using the WSNet simulator [76]. Since by construction, coverage is always provided and reading frequency respected, we mainly focus on the quality of IN load balancing in the simulations. Fig. 2.3(e) shows as example, the evolution of the zones distribution (represented by different grayscale) among 9 INs at different simulation times, when 250 TNs with random walk mobility have to be covered. Full details on the related algorithms and performance evaluation are available in [30].

2.3 Conclusion of the chapter

This chapter was dedicated to our contributions on the network-level services related to topology management. In this context, we have presented our contributions in terms of protocols and systems design for topology management through node adaptation and controlled mobility. The works related to node adaptation, i.e., SAND and NetGeoS, were based on flat solutions and on the idea of empowering sensor nodes with the ability to allow a specific structure to emerge from scratch while guaranteeing the good execution of a certain network functionality (e.g., routing or sensing fidelity). The presented works related to controlled mobility, i.e., Hilbert-based trajectory design and *Cover*, include flat and two-tier solutions respectively, and were designed for improving the efficiency of zone coverage by taking profit of the nodes mobility. In *Cover*, in particular, theoretical analysis and extensive simulations showed that the number of mobile target nodes per infrastructure node is close to the optimum at any time independently of the mobility pattern of the target nodes.

Although dealing with topology management through different approaches, the discussed contributions were performed according to common design principles: simplicity, localized knowledge, autonomous and distributed behavior. In particular, on-line deployment with limited overhead were always a main concern in the approaches, where nodes were supposed to use a small amount of information (e.g. only neighborhood discovery) and to generate a low processing overhead to take their activity decisions. After evaluation, the proposed solutions showed to efficiently reach the envisaged performance goals under the considered scenarios and applications. It is worth mentioning, however, that some assumptions need to be relaxed: e.g., constant speed of mobile nodes, uniform distribution of sensor nodes, to cite a few. In the *Cover* work, some of these issues are being considered in an extended version of the work (as the delegation of more than one cell for convergence speed improvement and the use of more realistic communication range among infrastructure nodes). Although no extension is foreseen for the other works, learnt lessons and gained experiences on their design have always indirectly influenced my further research activities.

3 Adaptive data management services

In general WSONs, but especially in wireless sensor networks, data management turns to be an essential design problem and must address challenges such as coordinating actions with other nodes and storage capacity. In the following, we present our adaptive solutions for the tasks related to data collection and data dissemination.

3.1 Supporting data collection

The most markedly function of WSNs is the possibility of perceiving what takes place in the physical world in ways not previously possible. Thus, although being data an important element in any computer network, it is in the context of WSNs that it becomes the central point of any protocol design. In particular, data collection strategies guarantee the gathering of the sensed data by special nodes, called sinks. We consider here scenarios of WSNs in which the sink node is mobile. The presence of mobile sinks that can directly collect data from sensor nodes in a monitored area avoids the necessity for node-to-sink path maintenance in the network and makes the network free to self-organize and to better respond to changing conditions.

In this context, data collection is related to how the mobile sink gathers the monitored data made available by the sensors in the network. In other words, depending on how data is stored in the network, the sink may adapt its trajectory, by either following a predetermined trajectory with controlled mobility (e.g., the sink must visit some predefined nodes to retrieve a representative view of the monitored area) [77] or move freely in the network following an uncontrolled mobility pattern and still harvest a representative view as long as it visits a minimum number of nodes, no matter which one. Data storage in the network for later collection may be characterized as *proactive*, if the monitored data is proactively distributed and stored throughout the network for being later retrieved by the mobile sink; or *reactive*, if data is sent towards the mobile sink as a reaction for the detection of sink's presence or queries.

The contributions presented in this section focus on proactive data storage strategies considering sinks with uncontrolled mobility pattern. A key challenge in this context is *how to effectively distribute and store monitored data such that it may be retrieved later by the mobile sink that freely determine its own trajectory while traversing the network.* We intend to limit the load in each individual storage node and to enable each node to determine its own group of storage nodes independently of other nodes without any implication on the randomness of each group. Hence, the solutions we had looked for need to replicate stored data in well selected storage nodes in a well balanced manner. This should enable the mobile sink to gather a representative view of the monitored region covered by n sensor nodes by only visiting a limited set of *any* m nodes, where $m << n$. To the best of our knowledge, we were the first to provide solutions for such challenges. Additionally, we have also investigated and analyzed the trade-off between the performance in data gathering and the performance in communication overhead of different strategies of proactive data dissemination that allow the mobile sink to move in an uncontrolled way to harvest the monitored data.

To achieve this, we have investigated different approaches: a Random Walk-based approach [51], the *DEEP* [32, 78], and the *Supple* [33] protocols. Such works are related to flat networking support solutions, periodical and explicit feedback adaptive service. They do not require a priori knowledge of all network nodes, do not use multi-hop routing, or any mechanism to track the sink. Finally, they improve data availability by replicating aggregated data in selected storage nodes in the network. In all studied approaches, each node may act as a storage node for some other nodes in the WSN, but not for all of them. The set of information that each node stores for other nodes (including for itself) is referred to as *partial view*, of size s. By slight abuse of terminology, we interchangeably use the term partial view both for the actual information stored at a given node p and for the IDs of the nodes whose information is stored at p. Our goal is to ensure that the partial view of each node will correspond to a uniform sample of the entire information (or entire set of nodes) in the system. This way, whenever the mobile sink visits a given node, it can collect a uniform sample of the information of all nodes in the system. At the following, I describe the *DEEP* and *Supple* approaches.

3.1.1 Density-based proactive data distribution protocol

(Publication [32, 78]. In collaboration with M. Vecchioa, A. Ziviani, and R. Friedman)

This strategy is based on an efficient *variable probabilistic forwarding strategy combined with a fixed probabilistic storing strategy*. We denote the resulting mechanism *DEEP* (*Density-based proactivE data dissEmination Protocol*).

The variable probabilistic forwarding F is inherited from (is set according to) the RAPID–like dissemination mechanism [79], which employs a combination of density sensitive probabilistic forwarding with deterministic corrective measures. In *DEEP*, however, we decided not to such corrective measures due to its long latency and added communication cost. Essentially, the goal here is to ensure that there will be a predefined average number of (re)transmissions of each message in each neighborhood. This is obtained as follows: each node p that receives a message m for the first time, decides to rebroadcast m immediately with probability $F = min(1, \frac{\beta}{|N(p)|})$, where $N(p)$ is the one hop neighborhood of p. The parameter β is called the reliability factor and represents the average number of nodes in each one-hop neighborhood that retransmit each received message m.

On the other hand, the storing probability S is fixed. In particular, we couple the above described probabilistic forwarding strategy with a storing strategy by which every node that receives a message carrying the data sensed by a given node stores this data with probability S. As we target partial views of size $s = \sqrt{n}$, we have also set $S = \frac{\sqrt{n}}{n}$.

We study how specific values of β impact the obtained partial view's size for both sparse ($d_{avg} = 6$) and dense ($d_{avg} = 24$) network classes. The experiments have been done using a discrete event simulator implemented in Matlab, with a simplified MAC layer, which includes message re-transmission (up to 6 retransmissions) and timeouts that are triggered when unicast messages are lost (we have adopted loss rates equal to 5% and 10% for sparse and dense topologies, respectively). Results clearly show that the β parameter can be used as a tuning knob to set the desired resulting partial view's size distribution. Yet, in order to obtain an average partial view's size of $s = \sqrt{n}$ (approximately 14 when the total number of nodes is 200), β should be set to 3.8 in sparse topologies and 5.8 in dense topologies so that the resulting partial view's size distribution in each case matches the target average.

The communication complexity of this strategy depends on the network density and is determined by the number of retransmissions each node performs. This results in a total communication complexity of $(F \times n) \times n = O(\frac{n^2\beta}{|N(p)|})$.

As a reference point for comparison, we have used the a random walk-based approach [51, 80], which ensure uniform distribution of the data. We have analyzed their data gathering efficiency, communication message overhead, and data distribution quality. We have shown that while the random walk-based approach is somewhat better than *DEEP* in terms of data distribution, the actual data gathering efficiency of *DEEP* is very close to the one provided by the random walk. Coupled with the fact that the probabilistic data dissemination mechanism of *DEEP* is much more efficient than the random walk-based approach in terms of communication overhead (i.e., *DEEP* generates about half less overhead), we conclude that *DEEP* is the more viable solution among the two. This is especially true for sparse networks, when the frequency of sending messages is low, and when the amount of sensed data reported in each message is large.

3.1.2 Flexible probabilistic data dissemination protocol

(Publication [33]. In collaboration with T. Herault, T. Largillier, S. Peyronnet, and F. Zaïdi)

This section presents our flexible proactive data dissemination protocol, called *Supple*. Supple is based on three phases described hereafter: neighborhood discovery, weight distribution, and data dissemination.

Tree-construction. In the first phase, Supple relies on a tree construction: A tree-based routing structure $T(G)$ initiated by a central-localized node in the network and that is at least binary. Let n be the number of nodes in the tree $T(G)$. The constructed tree $T(G)$ embeds the connectivity of the network and ensures that sampling a node according to a given distribution can be done with a logarithmic number of hops. In particular, Supple requires a bootstrap phase where $T(G)$ is constructed using a cost metric propagated in 1-hop `Hello` messages, as in [81]. The constructed $T(G)$ structure is

Table 3.1: Comparison between Random walk-based, DEEP, Supple, and flooding approaches.

	# rounds	msgs per round	total msgs	mem. overhead	additional overhead
RW-based	$r(s)$	$\frac{n^2}{2}$	$\frac{n^2}{2} \cdot r(s)$	partial view's size s	mem. for RW
Supple	$r(s)$	$n \cdot \log n$	$n \cdot r(s) \cdot \log n$	partial view's size s	3 integers per node
DEEP	1	$\frac{n^2\beta}{\lvert N(p)\rvert}$ broadcasts	$\frac{n^2\beta}{\lvert N(p)\rvert}$ broadcasts	partial view's size s	mem. for flooding
Flooding	1	n^2 broadcasts	n^2 broadcasts	n	mem. for flooding

thus, an aggregation of the shortest paths from each sensor to the central-localized node based on a cost metric, which can represent any application requirement: hop count, loss, delay among others. Finally, Supple can be adaptable to any kind of structure, the only requirement being the routing capability, such as: Peernet [82] and Tribe [5]. Sensors use such a simple tree-based structure to perform weight distribution among nodes.

Weight distribution. The second phase of Supple is the weight distribution. The flexibility of Supple is given by the fact that the data distribution can be adapted to any target set (denoted by S). For this, each node is assigned to a parameter $W(i)$ that defines its weights in the network. Weights are initially assigned to nodes based on an external criterion of storing nodes' selection. For uniform selection, all sensors will have the same weight and then, the same chances to be selected as storing node by another node, e.g., $W(i) = 1$. On the other hand, if the criterion is a location-based selection only nodes at the specified location will be used as storing nodes. For instance, if the criterion is a border selection strategy, each sensor i located on the border of the network will have $W(i) = 1$ and $W(i) = 0$ if it is an insider node .

The probability of sending data to a particular node i is given by the weight assigned to node i with respect to the sum of all weights of storing nodes in the target set S. Once local weights are assigned, nodes perform the weight distribution over the tree. The idea here is to initialize each node $i \in S$ with a triple $(l_i, W(i), r_i)$, where $W(i)$ is the weight of the node i in the target set and l_i (resp. third component r_i) is the weight of the left (resp. right) subtree of i: i.e., for all left child j of i, $l_i = l_j + W(j) + r_j$ (resp. for all right child p of i, $r_i = l_p + W(p) + r_p$). The complexity of the whole weight distribution process is $\theta(n)$. Additionally, the weight distribution only requires a field of at most $\log n + \log |W|$ bits in the usual `Hello` packet. Finally, weights are used by sensors at the data distribution phase.

Data distribution. This phase ensures the properly data propagation at storing nodes. Considering that nodes have the same local weight, this phase has to ensure a uniform distribution of nodes' data among the target set. The idea is the following. First, all nodes must send their data to the root of the tree (i.e. the node that started the tree construction). When the root receives new data from one of its children, it propagates it $r(s)$ times to its children. In [33], we have proved that to obtain with high probability, a partial view of size s, $r(s) = n \ln(\frac{n}{n-s})$ messages must be sent if $s \neq n$, or $n \ln n$ if $s = n$.

The propagation by the root is done according to the weights of its left and right subtree and also to its own weight (in the case the root is also in the target set). Moreover, it must be noted that messages are forwarded asynchronously, i.e., there is no reason for the root to finish the $r(s)$ sequential data sending of a node to start sending data of another node. The forwarding process naturally stops when the message is received by a node whose left and right component of the triple equals to 0 (i.e., at the leaf level). At the end of the data distribution, all nodes of the target set will have, with high probability, a partial view of size s. This view is randomly composed by nodes' data distributed according to the weights of nodes in the target set. In the case weights are equal for all nodes, we naturally achieve partial view of size s with uniformly disseminated data.

The complexity in term of messages of the whole process is $O(n \cdot r(s) \cdot \log n)$: Each node sends its data to the root, which implies in $O(n \log n)$ messages per node, and the root propagates each data $r(s)$ times through the tree (from the root to the leaves). In this way, *Supple* outperforms the message complexity of the previously presented works. Table 3.1 summarizes such results. For the special case where the partial view's size s of nodes is limited to $s = \sqrt{n}$ (i.e., $s \neq n$), we get $r(s) \approx \frac{nk}{n-s} \approx \sqrt{n}$. This means that for relatively small view sizes, there is a very little chance of getting collisions and that by only contacting $m = \sqrt{n}$ nodes in the target set S a sink can get a representative view of the whole data in the network (i.e., $s * m = \sqrt{n} * \sqrt{n} = n$). The communication complexity equals then to $O(n \cdot \sqrt{n} \cdot \log n)$. In this way, Supple allows an efficient data dissemination with an exponential improvement of the number of messages compared to random walk-based [51, 80] approach and *DEEP*. For a full description and analysis of the *Supple* protocol, we invite the reader to refer to [33].

3.2 Supporting data dissemination and/or delivery

Data dissemination strategies determine how data is propagated to destinations. It can also be used to organize data in the network in order to be later gathered by a static or mobile entity, as the data distribution strategies described in the previous section. Contrarily to the organization goals of the previous discussed data distribution strategies, this section focus on information dissemination performed in the context of wireless social or interest-based networks. To this end, we describe hereafter the *FairMix* [35, 34, 49] and *VIP delegation* [36, 54] works, both relying on on-demand and explicit feedback adaptive services. The first one relates to a flat network support approach, where all nodes collaborate to a content-based distribution. On the other hand, the second work relates to a two-tier deployment solution, where the data management in the network is performed by some specially selected nodes.

Before detailing such contributions, I would like to mention our contributions related to data dissemination in cognitive radio networks, corresponding to the PhD work of M. Rehmini, co-supervised by S. Fdida and myself, and performed in-cooperation with H. Khalife. In particular, we designed and evaluated SURF, a distributed channel selection strategy for improving data dissemination in multi-hop cognitive radio ad-hoc networks (CRNs). SURF classifies the available channels on the basis of past PU activities and of the number of cognitive radio SUs using the channels, and selects to use the best one in terms of CR neighbors and lower PU activity. Full details on the SURF design are available in [83, 56] along with a detailed simulation evaluation using the NS-2 simulator.

3.2.1 Content-based network coding to match social interest similarities

(Publication [35, 34, 49]. In collaboration with G. Karbaschi, S. Martin and K. Al Agha)

It is well known that network coding allows a multicast to reach the *max-flow min-cut* capacity in a lossless wireline network [84]. Nowadays, there has been a recent interest in employing network coding in multi-hop wireless networks since it allows to exploit natural advantages offered by shared wireless medium (such as spatial diversity, broadcast nature of the medium and data redundancy) [85]. As widely confirmed, by coding information in intermediary relays, network coding allows to improve throughput, reliability, and robustness in comparison to traditional routing algorithms [86]. Indeed, since wireless links cause packets to be spread about in a probabilistic manner, there is no reason to restrict information to a path such as wireline networks. Rather, each node that is potentially a relay can encode the packets it receives and sends them out. In such an approach, the concept of routing can be broken and the challenge lies on how to mix the incoming packets. In this way, network coding can be classified into two classes: inter-session coding where coding is allowed among packets belonging to different session and intra-session coding in which coding is limited to packets belonging to the same session. The original works on network coding show how intra-session coding improves the throughput of both unicast and multicast session in a lossy wireless networks. However, it is also shown that in the case of existing multiple flows, intra-session coding is not necessarily optimal [87] and in general inter-session coding across the flows is needed to achieve optimal throughput [88]. In this work, we claim that the benefit of higher gained throughput in inter-session coding happens at the cost of unfairness among users which are waiting for different data flows. To the best of our knowledge, this work constitute the first effort in the literature in investigating this unfairness issue.

In random linear network coding [89] (the coding coefficients are chosen randomly from a finite size field − such as Galois Field), a whole block of K packets linearly mixed through coding at the intermediate nodes in the network, requires the reception of K linearly independent encoded packets at each destination in order to be decoded correctly. Therefore, each packet in the data block must wait the reception of the whole block before it can be decoded, even if not all the packets in the block belong to the same destination. Hence, no specific packet can be set apart from other packets. As a result, a destination that is waiting for a single urgent packet, should wait to receive large enough encoded packets to recover all packets that are coded together. This may lead to a large *unfair* decoding delay for small blocks in the presence of various block sizes. Looking from a service provision's perspective, it makes nodes demanding for few resources (e.g. few bandwidth – for destinations expecting few packets) being penalized by high consuming nodes. Considering multi-hop and static wireless applications which require transmitting heterogeneous files (e.g. video or music files) to dmultiple destinations, delay performance may be critical to the satisfaction of the users. In this case, it is essential and extremely challenging to deal with unfairness issues.

Our solution to alleviate the unfairness in decoding delay of inter-session network coding is called *FairMix*. FairMix separately mixes the packets of *any* source to *each* destination. Besides of enjoying from network coding advantages through mixing all the packets going to the same destination, FairMix aims to make a distinct for decoding delay of each destination corresponding to the size of the data blocks. We define the *decoding delay* as the difference between the time instant of sending the first packet of a block and the time instant of receiving the last packet of the block at each specific destination. This shows the completion time for transmitting one block to its destination.

FairMix can be interpreted as a *fair* intra-session network coding where each session is identified by its end-point destination. Therefore, each node N_i maintains a virtual queue per destination d_j, denoted as $\mathcal{V}_{d_j}$, which contains packets at N_i which are destined to d_j. Once N_i has a transmission opportunity it chooses a queue according to scheduling policy and generates an encoded packet across all the packets in that queue (the impact of different scheduling policy on the FairMix performance has been explored in [49]). As a result, throughout the network the linear equations of packets of each destination are treated independently from other destinations. Hence, in FairMix, a destination d_i can decode a whole block of $K_{d_i} = \sum_{j=1}^{|s|} K_{s_j}^{d_i}$ packets upon receiving K_{d_i} (instead of K) independent encoded packets from *any* relay nodes.

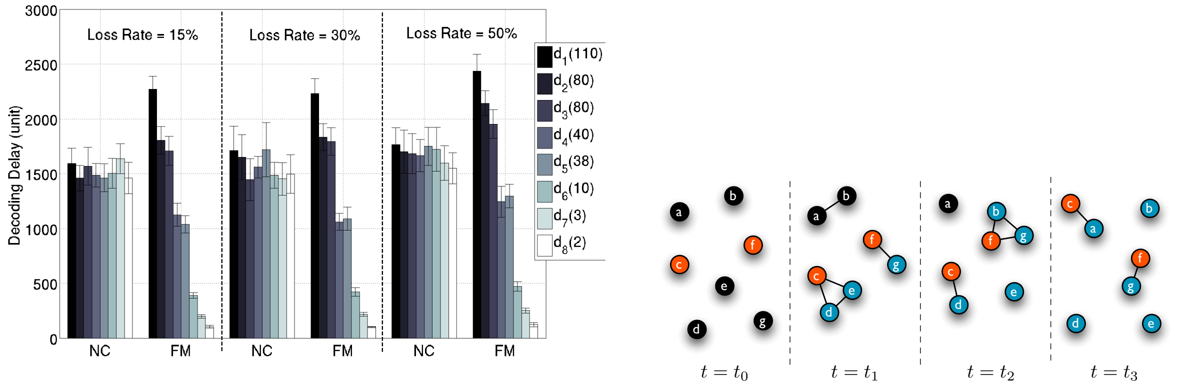
The decoding of data blocks is done at the destination nodes. Therefore, at each packet reception, if a receiving node N is a destination, say d_i, it verifies if it is able to decode its whole data block, i.e., if it has received the $K_{s_i}^{d_i}$ innovative packets. If so, the decoded packets are delivered to the application layer and an ack packet $ack(d_i, I)$ is scheduled for transmission, according to any stopping mechanism (e.g., as Immune or Vaccine protocols) being considered.

Using a custom time-based network coding simulator [90], we have studied and compared by simulations the delay and fairness performance of FairMix and naive inter-session network coding, when applied in a multi-hop and static wireless network [35, 49]. We evaluated the impact of: (1) data block size, (2) number of sources, (3) scheduling policy, (4) finite field size in which coding symbols are generated, and (5) forwarding mechanisms. We showed the benefit of FairMix in terms of fairness, specially in the case that data block sizes destined to different destinations are not the same. Moreover, FairMix provided considerable delay gains compared to naive network coding for destinations waiting for small block sizes (see Fig. 3.1(a)). This last point is particularly interesting in providing different quality of services with differentiated priorities since servicing the high priority traffic (such as urgent messages) separately from ordinary traffics can be easily envisioned. Additionally, FairMix could be used as a content-based network coding that aims to match the social interests similarities (i.e., same profession, hobbies, interests, etc.) of people in a community. In this way, besides lower decoding delay, FairMix provides a more adaptive-to-social-network coding for the users that are interested in different contents.

3.2.2 Offloading data in wireless social mobile networks

(Publication [36, 54]. In collaboration with M. V. Barbera, J. Stefa, M. D. de Amorim, and M. Boc)

Since the modern smart-phones have been introduced worldwide, more and more users have become eager to engage with social applications and connected services. In such context, opportunistic wireless *social ad hoc networks* are emerging a new alternative to provide wireless data access in urban areas. These networks allow content sharing between wireless users without requiring any pre-existing Internet infrastructure, by simply relying on short-range opportunistic communications (Bluetooth or Wi-Fi) between wireless devices whenever possible. Nevertheless, communication in such opportunistic wireless ad-hoc networks is challenging due to the volatility of contacts and is strongly impacted by human mobility, which is driven by user social behavior. In this way, the nature of human interaction makes data management in such networks challenging. Additionally, smart-phone owners are using an increasing



(a) FairMix delay gains.

(b) VIP selection example.

Figure 3.1: (a) FairMix (FM) and naive network coding (NC) evaluation in terms of decoding delay in a 100-node network with four randomly placed sources, varying loss rates and block sizes (from 2 to 110 packets). (b) VIPs c and f cover the network after three time stamps.

number of applications requiring the transfer of large amounts of data to/from mobile devices.

We proposed *VIP delegation*, a solution to these problems based solely on the inherent social aspects of user mobility. Our idea is to exploit a *few, important* users, called *VIP nodes*, that with their movements and interactions are able to provide communication support to other users in the network. These VIP devices would act as a bridge between the network infrastructure and the remaining of the network, each time large amount of data has to be transferred. We focus here in delay tolerant applications. Some examples of such type of applications that could benefit by these VIPs: collection of urban- or participatory-sensing related data; distribution of large content to users by service/software providers (e.g., software updates and recurrent security patches); free update of mobile software's ad pools.

VIP delegation in a nutshell: The movement of smart-phone users is not random; rather, it is a manifestation of their social behavior [91]. This movement, along with wireless interactions among users, generates a social mobile network. The analysis of such mobility patterns and the understanding of how mobile users "interact" (i.e., meet) play a critical role at the design of solutions/services for such kind of networks. In general lines, this work investigated the following questions: (i) *how to gain insights into social mobile networking scenarios* and (ii) *how to utilize such insights to design solutions allowing data management support in opportunistic mobile ad hoc networks.*

In particular, though the number of network users can be very large, just a few of them have an "important" role within the social graph induced by the encounters. The natural behavior of these VIP nodes, which are considerably fewer than the rest of the network, can be a valuable resource in both information dissemination and collection to/from the rest of the network. Motivated by the fact that opportunities for users to exchange data depend on their habits and mobility patterns, our idea is the following: Turn those *few* VIP nodes into bridges between regular users and the Internet, each time large amount of data is to be uploaded/downloaded by these latter ones. In a word, the VIP would act as delegates of the network infrastructure builder. When a VIP visits a user, we say that the user is covered. Fig. 3.1(b) illustrates a network example at three different times: nodes {c,f} can serve as VIPs on behalf of the others, as they have met by themselves from $t - t_0$ to $t = t_3$ the rest of the network.

Now the problem becomes the following: *how to choose the smallest VIP set that with their natural movement in the network cover potentially all users daily?* As described in the following, we solve this problem by presenting two VIP selection methods that rely on either a global or a local view of the network.

Training period and social graph: The computation of VIP sets in a social mobile network requires the so called *network social graph.* For this, we have observed users' movement and meeting patterns for a short period of *1 week.* The length of such *training period* is not casual. Usually, our life and the activities we conduct are organized on a week-base, mostly having a common routine repeated day by day. Such repetition also infers the common meetings generated by those activities. Hence, we were able to generate a *social graph* $G(V; E)$, where V is the set of users and E is the set of social ties among them. Social ties (edges) in the graph were strictly related to users' contact frequency: A link exists between two users if the number of times they meet is higher than a certain threshold, during the training period.

VIP selection methods: We defined the importance of a node in the network by applying to the network social graph several well-known structural attributes in social network theory: betweenness centrality, closeness centrality, degree centrality, and page rank (detailed in [36, 54]). We then proposed, formalized, and evaluated two methods of VIPs selection:

- The **global VIP delegation** aims to select the smallest VIP set over the global social graph that is able to daily cover the network through direct contacts with network users. For this, the nodes are first ordered according to each of the earlier described social metrics, and then one of the following VIP promoting ways is applied: (1) *blind global promotion* selects the top-ranked nodes not yet promoted, till the network is covered; (2) *greedy global promotion* is a set-cover flavored solution: The top-ranked nodes is promoted, nodes it covers are dumped and rankings on the remaining nodes are re-computed. This is repeated till the network is covered.
- The **neighborhood VIP delegation** selects users that are important within their social communities. We first detect social-communities using the k-clique community algorithm. Afterwards, nodes are ranked according to the described social metrics. Then we start covering each community by promoting its members to VIPs similarly to the global VIPs methods: (1) *blind hood promotion* continuously selects the top ranked nodes not yet promoted in the community, till the network is covered; (2) *greedy hood promotion* is a set-cover flavored solution applied to each community: The highest-ranked member in the community is promoted, nodes it covers within the community are dropped, and rankings are computed again in the remaining community graph.

Benchmark approach: In order to evaluate the efficiency of our strategies, we proposed a benchmark approach that gives the optimal solution: 100% of user coverage daily, with minimum number of VIPs. This benchmark serves only for comparison purposes, as it requires knowing the future to compute the exact set of VIPs. It is obtained by abstracting our application scenario to a formal representation and by finding a minimum out-dominating set under L-reduction (a Set Cover solution for the NP-hard problem of the Minimum Dominating Set).

We evaluated the performance of the global and neighborhood VIP delegation methods in terms of network coverage, by varying the number of VIPs chosen. For this, we use two real data-sets: Dartmouth [92] (movement of students and staff in campus) and Taxis [93] (movement of cabs in San Francisco)– and three synthetic datasets of the Cambridge Campus data-set generated with the SWIM mobility model [94]. We compare our solution with an optimal benchmark computed from the full knowledge of the system. The results reveal that our strategies get very close to the performance of the benchmark VIPs: Only 5.93% page-rank VIPs against almost 4% of the benchmark's set to offload about 90% of the network in campus-like scenarios. Moreover, contrarily to the benchmark approach where a complete knowledge of encounters in the network is required, all our methods rely on a short network observation period of 1 week, and select VIP sets that result small, efficient, and stable in time. In [36, 54], a full description of strategies and analysis can be found.

3.3 Conclusion of the chapter

This chapter presented our contributions dealing with adaptive data management in wireless self-organizing networks, where flat and two-tier networking support solutions were designed for data collection and dissemination in such networks. In all the presented works, mechanisms prioritizing an autonomous and distributed behavior only based on localized network information were considered.

The presented flat-based works providing support to data collection, i.e., *DEEP*, and *Supple*, have focused on decoupling the data dissemination management from the mobile sink's trajectory by uniformly distributing data through well selected storing nodes in the network. *DEEP* and *Supple* approaches showed to be much more efficient than the Random-Walk approach [51, 80] in terms of communication overhead, while ensuring a data gathering efficiency very close to the best one provided by the Random-Walk [51, 80]. In particular, *Supple* provides an exponential improvement of the number of messages used, when compared to Random-walk and *DEEP* approaches. The choice to use a tree, however, introduces a high overhead of messages transmission to the root and its vicinity. In order to deal with this drawback, another of our on-going work named ProFlex considers the construction of multiple data replication structures, which are managed by more powerful nodes [46]. Due to page limitation, this work is not detailed here.

It is important to mention here that clearly, if low latency is not required by the application (e.g., in continuous monitoring), the data stored in partial views as well as data transmitted between nodes can be compressed (e.g., the algorithm in [95] reaches compression ratios up to 70% on environmental datasets). This would reduce the network traffic and thus, prolonging the network lifetime. Compression algorithms are, however, related to data processing task, which is orthogonal to our works.

In a second part of this chapter, the *FairMix* and *VIP delegation* contributions related to data delivery in the context of wireless social networks were introduced. Contrarily to the previous discussed approaches, such works take into account the characteristics of the network social structure. This concerns performing dissemination according to the social interests' similarities of people or groups (or communities) in static social networks (i.e., in *FairMix*) or exploring the inherent social aspects of movements and wireless interactions of users in mobile networks (i.e., in *VIP delegation*). Although to be in an initial stage, such contributions allowed us to identify some interesting open research directions. We discuss some of them in the following.

In the VIP delegation case, nodes selected as VIPs are supposed to have sufficient resources in both battery and memory. To account for this issue, two propositions could be considered: (i) taking into account the users' traffic load at the delegates selection and use it to establish a maximum load threshold per delegate and/or (ii) nodes selected as delegates could have their devices upgraded for "working" for the network provider/application builder (considering the quite low number of selected VIPs by our strategies to guarantee 90% coverage: 8% in SWIM, 5.93% in Dartmouth, less than 1% in Taxi). Additionally, two-hop forwarding could be added to enhance the coverage and required numbers of VIPs provided by the strategies. For instance, the traffic of 10% of network users that remain uncovered by our delegates could be forwarded to two-hop distant VIPs, taking profit of the transitivity aspect (the probability that two users have common friends) of human social networks.

Finally, concerning the *FairMix* work, our next steps include considering the possibility of tailoring network coding for information dissemination in social mobile networks. The limited resources of hand-held devices and their dense distribution prevent the use of pure-based flooding techniques, and suggest the design of more efficient data propagation techniques. To this end, community detection and social network analysis are the tools to help understanding the social structure of a community and thus improving the information dissemination among the users. We believe the propagation benefits introduced by network coding coupled with solutions able to detect and to adapt to connectivity variations in the network, i.e., to nodes' social interactions as well as to different content distribution paradigms, may result in interesting dissemination approaches providing reliable and robust data delivery. These approaches would dictate the behavior of a coding node in terms of *"when and what to code"*.

4 Adaptive Routing and Forwarding services

In this chapter, we discuss the last remaining network-level service that drove my research work, which is related to routing and forwarding services. The factors that drive the adaptation in these services are: mobility, connectivity opportunity in scenarios with intermittent connectivity, and energy consumption.

My interest in this type of services started during my PhD thesis, when I designed adaptive routing solutions providing scalability support in WSONs. Subsequently in 2007, in [48, 47] (PhD thesis of J. Rahme, co-supervised by K. Al Agha and myself), we have considered adaptive routing solutions to address energy consumption in WSNs. Nevertheless, because of space limitation, in the following section, I decided to describe our adaptive forwarding works addressing connectivity opportunity and mobility in delay tolerant networks. Such works (1) rely on on-demand and explicit feedback adaptive services and (2) relate to flat networking support solutions. Before describing the designed adaptive protocols, I first discuss the principles related to adaptive forwarding.

4.1 Adaptive forwarding principles

(Publication [52]. In collaboration with C. Sengul and R. Friedman)

Adaptive forwarding aims at achieving high reliability and timely delivery with an acceptable overhead under different conditions. Clearly, these goals can be contradicting. The simplest way to achieve high reliability and timely delivery is epidemic routing [96]. However, the good performance comes with a high cost. Furthermore, if buffer and bandwidth constraints are taken into account, this performance might not be realized in practice. For instance, in a highly dense network with low mobility, epidemic routing might incur too much overhead, degrading also reliability. Another immediate solution that comes to mind to reduce the overhead is to always relay through the node that is likely to meet the destination first. Nevertheless, keeping track of and disseminating all nodes that might eventually meet the destination either directly or indirectly might be too costly. Furthermore, approaches that try to reduce the cost by selecting a few but good relays, for instance, by comparing the last meeting time with the destination or contact frequency [97], might be too conservative and lose good forwarding opportunities in environments with scarce connectivity. A less reserved approach that optimizes resource consumption might require a control channel to maintain, for instance, the number of and location of replicas [98]. Hence, to strike a balance, a protocol must be able to adapt.

Adaptive forwarding requires matching forwarding decisions to different network dynamics that occur due to low node density, non-deterministic node mobility, or frequent end-to-end disconnections. Additionally, when there are no disruptions due to mobility, it should be possible to take advantage of this and incur lower costs for higher reliability. Therefore, nodes need to learn from past observations: *if a pattern exists among past contacts, future contacts can be estimated.* Such estimations would be invaluable for managing message buffering and transmissions.

On the flip side, when predictions do not match future behavior, it is necessary to rewind bad forwarding decisions to improve performance. Nodes should also operate in an opportunistic manner and take advantages of new meetings. Hence, adaptive forwarding should not be too tied to predictions. By rewinding decisions, nodes should be able to save messages, based on local observations and through well-designed buffer management mechanisms. We assert this flexibility is the key to adaptive forwarding.

To counter these challenges, we present hereafter the *Seeker* [41] and the *GrAnt* [42, 55, 13] protocols, two *adaptive forwarding approaches that match forwarding decisions to different mobility and operating conditions.* For this, both protocols provide mechanisms to gather from network dynamics, information describing how promising nodes are as relays. While *Seeker* is based on probabilistic forwarding decisions, *GrAnt* suggests the use of deterministic decision rules.

Our network model assumes that nodes may exhibit different levels of mobility, including being disconnected at certain times. Nodes, including destinations, might be static or mobile and might have limited buffer or bandwidth resources. Each node is able to communicate with a subset of neighbors that are in its transmission range. We do not assume symmetric communication. Although it does not affect correctness of the protocols' operation, communication symmetry or duration of contact improves the contact quality estimation. Nodes do not know their location or any topological information, such as where the destinations are. Still, our objective is to design protocols that correctly propagate messages to destinations, thereby achieving high reliability and timely delivery with an acceptable overhead.

4.2 *Seeker*: a history-based randomized forwarding protocol

(Publication [41]. In collaboration with C. Sengul, R. Friedman, A.-M. Kermarrec, and M. Bertier)

Intuitively, *Seeker* operates as follows. When a node j receives messages from node i destined to node m, j tosses a biased coin to decide whether to forward these messages or not (*receiver-based forwarding*). The weight of the coin is based on the expected connectivity to m and corresponding connection quality. That is, the more j believes it is on a good path leading to m, the more likely j is to carry and forward the message for i. However, as typically connectivity and connection quality cannot be known ahead, *Seeker* needs to make estimations and be able to recover from bad decisions. *Seeker* incorporates the following four mechanisms to achieve adaptivity.

Contact history maintenance. *Seeker* decides which nodes may be promising relays observing network dynamics. To this end, forwarding decisions are based on contact quality, which captures not only contact frequency between two nodes [97] but also their communication quality (i.e., whether the nodes agreed to carry messages for each other). This affects how long a message is buffered and allows *Seeker* to consider different bandwidth constraints and operating conditions of nodes. Connectivity and contact quality information is maintained at each node in a *contact history*: A node builds a contact history as it meets other nodes in the network. Connections are detected by the reception of messages, while nodes detect disconnections by monitoring silent periods. A node discovers a new contact when it receives a message from that node. *Seeker* utilizes four types of messages: (1) the *Data* message to be sent to its destination; (2) the *Hello* messages, which are periodically sent; (3) the *Ack* message, which is sent to the previous hop by the destination *only* to confirm that the destination received the *Data* message and to validate the connectivity quality to the destination for the previous hop node; (4) the *Promise* message only indicates a contact's willingness to forward the message (but does not guarantee it) and it is sent by the next hop to the previous hop only.

As nodes detect connections and disconnections, they update their tables accordingly. In case the contact history table is full, the new contact might replace an old contact with low quality value. Finally, by not limiting the contact information to destinations, nodes are able to exploit *transitive contacts*. This information is essential for nodes that never come into contact with a destination.

Estimating contact quality. Based on the contact history, nodes make simple *estimations about future meetings* with their contacts. To evaluate contact quality, each node i calculates (1) the estimated remaining time to meet a contact j, $t^i_{j,wait}$ and (2) the expected *service quality* from the contact j, $c^i_{j,m}$ (i.e., the rate j responds with *promise* messages to *data* messages for destination m). Then, using $t^i_{j,wait}$ and $c^i_{j,m}$, a node i calculates a quality value for each neighbor j for a given destination m, denoted $q^i_{j,m}$, as follows: $q^i_{j,m} = e^{-\frac{t^i_{j,wait}}{t^i_j(k+1))^P}} \cdot c^i_{j,m}$. The estimated remaining time $t^i_{j,wait}$ equals $t^i_{j,last} + t^i_j(k+1))^P - t$ if j is not in contact with i, and 0 otherwise. Its normalization is performed in order to adapt to varying time granularity with different mobility patterns, where $(t^i_j(k+1))^P$ is the expected inter-contact meeting time between i and j computed using an exponentially weighted moving average

Adaptive and opportunistic forwarding. Contact quality estimations drive the adaptive and opportunistic forwarding mechanism. Depending on their path quality to the destination of a message, nodes only forward the message (in a broadcast mode) at a time when they expect to meet their best contact for it. The adaptive forwarding mechanism decides if and when a message should be forwarded. In particular, nodes in *Seeker* toss a biased coin (dictated by p^i), to make forwarding decisions: $p^i = \max\{p^i_{max}, {p^i_{avg}}^{\frac{1}{n}}\}$, where for all j, $p^i_{max} = \max q^i_{j,m}$, $p^i_{avg} = \sum_j q^i_{j,m}/n$ and n is the number of contacts. The goal here is (1) to emphasize both quality and the number of contacts when computing p^i and (2) to introduce some randomness in forwarding, which helps learning and adaptation, especially when there is not enough information on contact quality.

Nodes thus adapt their forwarding decisions, i.e. "forward" or "not forward", to the current conditions. If the decision is "not forward", the message is still buffered for a short time to allow reevaluating decisions and saving messages. If the decision is "forward", the message is buffered until the next meeting time of the best recorded contact (e.g., due to more frequent sightings of the contact or better communication performance). When this time comes, nodes locally broadcast the message to take advantage of other nodes in the vicinity. As the message does not identify a next hop node, the responsibility of forwarding the data is left to its receivers. Therefore, even if the message is not received by the expected contact, it may continue its probabilistic propagation based on the node degree at the message sending time. Our goal is to utilize the existent bias among different contacts for more reliable and resource-efficient communication. Hence, messages are propagated to destinations in a headlight manner.

Seeker allows *rewinding forwarding* decisions based on the current observations to improve performance and cost. Nodes reevaluate their forwarding decisions as follows: (1) the nodes that are in the vicinity of the destination and hear the destination send an *Ack*, cancel the transmission of the message; (2) if a node hears a threshold number of *Promise* messages for a data message, it reduces the number of copies, x, it has to send by one. While rewinding a forward decision improves cost, rewinding a "not forward" decision might improve performance. A node changes its "not forward" decision if it observes no response for a message (i.e., a *Promise* or an *Ack*).

Buffer management. Buffer management in *Seeker* becomes important as it allows storing data in the expectation of meeting a good contact and rewinding some forwarding decisions. Two different buffers are used. *Send* and *Quarantine buffer*. The *send buffer* stores messages that are waiting to be sent: If the node chooses to meet a better contact, the message is buffered until the meeting time of that contact. If a node decides not to forward a message, it is stored for a short time in *quarantine buffer*. When deleting messages from send buffer, we use the *source prioritized drophead* policy, where a node drops the oldest relay messages first followed by the oldest source messages. A simpler policy of dropping the oldest messages first is applied to the quarantine buffer.

Full details on the design of the *Seeker* protocol is available in [52, 41] along with a simulation evaluation using the NS-2 simulator. Using real mobility traces (i.e., MIT Reality Miningÿ[99] – scarce connectivity and longer contact duration – and Haggle Cambridge [100] – plenty connectivity and shorter contact duration), we show that *Seeker* is able to adapt its forwarding accordingly in diverse scenarios (i.e., with different patterns of connectivity and static scenarios) and achieves high performance with lower overhead. The *Seeker* performance was compared to Prophet and Delegation forwarding approaches as well as to DSDV protocol (i.e., in the static scenario). *Seeker* does not consider infinite buffers or bandwidth and allows nodes to modify their forwarding strategy based on their perception of the past and current conditions. Hence, it achieves high flexibility, which is invaluable for DTNs.

4.3 *GrAnt*: a GReedy ANT Colony Optimization-based forwarding protocol

(Publication [42, 55, 13]. In collaboration with A. C. K. Vendramin, A. Munaretto and M. R. Delgado)

In GrAnt, we are investigating the use of Ant Colony Optimization (ACO) in DTNs. *Ant Colony Optimization* (ACO) metaheuristic is an example of an artificial swarm intelligence system which is inspired by the collective behavior of social insects [101]. In ACO algorithms, usually an artificial ant collects information about a problem, stochastically makes its own decision, and constructs solutions in a stepwise way. The behavior that emerges is a group of relatively "not intelligent" ants that interact through simple rules and dynamically self-organize maintaining their positions around the shortest trails: Ants leave their nest without information about the location of food sources, move randomly at initial steps, and deposit a substance called pheromone on the ground. The pheromone marks a trail, representing a solution for the problem, that will be positively increased to become more attractive in subsequent iterations and to serve as a history of the previous movement of the best ants. The greedy ACO-based forwarding mechanism implemented by *GrAnt* directs the traffic to the most promising nodes with the aim of improving the messages delivery while limiting the message replications and droppings. The term greedy suggests the use of deterministic decision rules, where the best forwarder option is always chosen, instead of the probabilistic ones commonly used in the ACO paradigm. Decision rules consider heuristic functions and pheromone concentration.

GrAnt brings schemes for: gathering updates from network dynamics, determining the best paths to be followed to a message reach its destination, scheduling message transmission (i.e., decides the order

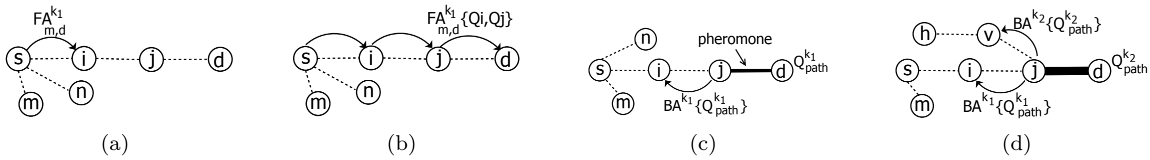


Figure 4.1: Overview of the GrAnt protocol execution.

in which messages are transmitted), and managing buffer space (i.e., indicates which messages can be discarded from the buffer when it reaches its occupancy limit). At the following, I describe the two first schemes and invite the reader to refer to [42, 55, 13] for details about the full protocol.

Learning from network dynamics. To direct the DTN traffic to the most promising contacts, *GrAnt* uses information about opportunistic social connectivity between nodes. It characterizes the utility of each node as a message forwarder, by considering: its accumulated degree centrality, its betweenness utility, and its social proximity with other nodes. In particular, a exponentially weighted moving average is used to predict the nodes' future degree centrality, i.e., $DC_i(t + \Delta t)$. As we are interested in different paths to each destination, we compute the nodes' betweenness utility in a different way than traditional betweenness centrality. We define our betweenness utility, i.e., ($BetwU_{i,d}$), as the frequency that a node i appears in paths between any source node and d. Finally, the social proximity metric among two nodes, i.e., $Social_{i,j}$, considers the prediction of their future contact duration (i.e., $d_{i,j}(t + \Delta t)$) combined with their contacts frequency (i.e. $\lambda_{i,j}$, the number of times i and j established a contact over the time window T): i.e., $Social_{i,j} = \frac{\lambda_{i,j} \times d_{i,j}(t+\Delta t)}{T}$.

Determining the best paths to destinations. The *GrAnt* routing module falls under the category of prediction-based protocols as it observes the nodes' behavioral patterns to ensure: a good message delivery rate, fewer dropped messages, and a low cost in terms of the transmitted message replicas. It is composed by a *path search phase* and a *backward phase*.

In general, the *GrAnt* protocol works as follows. Once a path search is requested in a source node, a small control message called *Forward Ant (FA)* is created, encapsulated into the data message and sent toward the destination d via one or more intermediate nodes (see Fig. 4.1(a)). The path to d is constructed based on the knowledge acquired by this FA, which dictates the forwarding decision at a node and tries to infer the capability of good next forwarders to d. The forwarding decision at a node i is performed by adopting a greedy transition rule, which considers two metrics very popular in ACO paradigms: the *pheromone* ($\tau_{(i,j),d}$) at link (i, j) in the path to a destination d and the *heuristic function* (i.e., $\eta_{(i),d} = BetwU_{i,d} + Social_{i,d}$) associated to a node i in the path to d. Additionally, it is worth mentioning the utility assigned to a node may change according to whether node i is the source or an intermediate node of a message. We invite the reader to refer to the pseudo-code described in [55, 13], for a full description on the forwarding decisions' conditions.

While being forwarded, each FA collects the degree centrality of every node composing the path to d (see Fig. 4.1(b)). Once the destination is reached, the total quality of the path found by the FA k, i.e., $Q_{path^k_{s,d}}(t)$, is calculated considering the average degree centrality of nodes in the path and the reciprocal of the number of hops composing it: i.e., $Q_{path^k_{s,d}}(t) = \frac{\sum_{\forall i \in path^k_{s,d}} DC_i(t+\Delta t)}{nHops} + \frac{1}{nHops}$. It is worth mentioning that smaller is the number of hops in a path, less network resource will be consumed and less interference will be generated.

The destination creates then, a *Backward Ant (BA)* message, storing the computed total quality of the path found by its corresponding FA. The BA is sent through the reverse path indicated by the FA. The reception of a BA k sent from a node j to each neighbor i produces two effects: (1) increases by one the node i's betweenness utility (i.e., $BetwU_{i,d}$) to the destination d of the message and (2) deposits a *pheromone* on the link (i, j) toward d (i.e., $\tau_{(i,j),d}$). The pheromone is used to mark a trail, which is proportional to the total quality at links between nodes composing the reverse path: i.e., $\tau_{(i,j),d}(t) = (1 - \rho) \times \tau_{(i,j),d}(t - \Delta t) + Q_{path^k_{s,d}}(t)$ if $i \in path^k_{s,d}$, where $\tau_{(i,j),d}(t - \Delta t)$ is the pheromone of link (i, j)

last updated, $Q_{path^k_{s,d}}(t)$ is the pheromone concentration deposited by the just received BA k on link (i, j), and ρ defines the pheromone evaporation rate (see Fig. 4.1(c)). The idea of using a reverse path is motivated by the fact that wireless social networks exhibit the small world phenomenon which comes from the observation that individuals are often linked by a short chain of acquaintances and that encounters are sufficient to build a connected relationship graph. Additionally, repetitive encounters between pairs of nodes and frequent visited locations by nodes can be also observed. Finally, if subsequent messages are forwarded to the same destination, the already deposited pheromone will be reinforced at those links and will help the forwarding of future FAs to the same destination (see Fig. 4.1(d)). Additionally, the BA serves as an acknowledgment of the message received by the destination, allowing the nodes which still maintain the message to discard it. A node that encounters another node that has already received a BA for a given message, will delete the corresponding message and its associated variables.

In summary, the adopted strategies by *GrAnt* allow the maintenance of a set of alternative paths to each destination. At the beginning, only lower quality contacts will be available, however, due to the dynamics of nodes in DTNs, after some time, a higher percentage of FAs will provide a faster discovery of new paths and/or the intensification of already existing ones.

To the best of our knowledge, *GrAnt* is the first unicast protocol that employs a greedy ACO which: (1) infers best promising forwarders from nodes' social connectivity, (2) determines the best paths to be followed to a message reach its destination, while limiting the message replications and droppings, (3) performs message transmission scheduling and buffer space management. Simulation results obtained using the ONE simulator show that in scenarios implementing two different mobility scenarios (i.e., one activity-based scenario (named Working Day) and another based on Points of Interest), GrAnt achieves higher delivery ratio, lower messages redundancy, and fewer dropped messages when compared to Epidemic and PROPHET. Full details on the metrics and algorithms related to the greedy ACO-based forwarding protocol are available in [55, 13] along with the simulation evaluation details.

4.4 Conclusion

Much research on delay-tolerant networks tries to address routing issues when there is no contemporaneous path between nodes. Such approaches, however, were designed focusing basically on the problem of intermittent connectivity caused by the constant mobility of users and consequently, do *not consider the shift of network environments or operation conditions, such as users with high to low or even no mobility.* Consequently, they present low adaptation capabilities and are therefore too conservative and often miss good forwarding opportunities, especially in environments with scarce connectivity. Our claim in this chapter is that *forwarding algorithms should not be designed by always assuming high mobility or long disconnections.* Hence, the main focus of our researches here was to adjust forwarding decisions on the fly while respecting resource constraints. In this context, we have introduced *Seeker* [52, 41] and *GrAnt* [42, 55, 13] protocols, which bring the ability to match different mobility patterns, under more realistic assumptions (i.e., it does not consider infinite buffers or bandwidth).

In addition, both protocols try to infer the most promising forwarders to destination with the aim of improving the messages delivery while limiting the message replications and droppings. While in *Seeker* the most promising forwarders are determined by observing contact frequency and communication quality (i.e, if nodes agreed to carry messages for each other) among nodes, *GrAnt* is more devoted to taking profit of human mobility pattern and to understanding social network properties. In particular, given that the adaptation in nature is a permanent and continuous process, and that as in biological networks most social wireless networks display substantial non-trivial topological features (with connectivity patterns that are neither purely regular nor purely random), *GrAnt* investigates the use of Ant Colony Optimization (ACO) in DTNs. The most promising paths to destinations are determined based on the previously deposited pheronome and on the social interactions' quality of nodes. *GrAnt* uses deterministic decision rules instead of the probabilistic ones as implemented by *Seeker* and commonly used in the ACO paradigm.

In summary, we have been considering adaptive forwarding protocols that (1) rely on social-based approaches or/and (2) that are able to deal with a shift of network environments or operation conditions of nodes. Additionally, the inspiration gotten from social structures and nature have been shown to be good directions for full adaptive forwarding algorithm design. Hence, we will more and more integrate such inspirations from the very beginning of the process design of our protocols.

5 Conclusions and perspectives

The research contributions discussed in this manuscript were guided by the main goal of providing network-level support for success data delivery in wireless self-organizing networks. The different types of wireless self-organizing networks require adaptive networking services targeted to deal with their dynamic nature (i.e., mobility, resource limitation, unreliable wireless communication, etc) and to find a fit between their operation and the environment. The research axes I developed together with my colleagues in this context are categorized in adaptive core and network-level services (as shown in Fig. 1.2). These two categories of adaptive services are distinguished by the level where such dynamics are considered, i.e., at the node or at the network level. The contributions related to core or node-level services that I performed relate to location service and neighborhood discovery services and were mentioned in the introduction of this manuscript. The remaining chapters of this manuscript were, however, devoted to the research that I conducted around adaptive network-level services. Therefore, I structured this manuscript in three main chapters corresponding to three classes of network-level services: topology management services, data management services, and routing and forwarding services.

In the first chapter, the contributions related to topology management services through node adaptation and through controlled mobility were presented. I first described the ones related to node adaptation, named *SAND* [27] and the systems *VINCOS* and *NetGeoS* [26, 44, 45, 57, 58]. These works deal with energy-conserving topology management or with geometric self-structuring of nodes in wireless sensor networks. By focusing on networked autonomous systems, such works provide simple solutions for empowering sensor nodes with the ability to form a network and to operate in a decentralized self-organized manner, while maintaining connectivity and performing area monitoring. Still in this chapter, I described the topology management solutions based on controlled mobility, named Hilbert-based trajectory design [31] and *Cover* [30] approaches. By taking profit of the moving capability of networking nodes, such solutions assure the adaptive coverage of a monitored zone, while considering application reading frequency requirements and environment changes. We will keep on working in some *Cover* improvements by investigating ways to reduce the number of infrastructure nodes and by proposing loose algorithms that allow some parts of the area to be temporarily uncovered if empty.

The second chapter presents my contributions on the data management support. In this context, I structured my activities into two main areas. The first concerns data collection, which involved data distribution solutions with organization goals. The question that conducted my research activities in this area was: "How to efficiently distribute and store in the network the data collected by n sensor in such a way that a mobile sink performing *any* type of trajectory can gather a representative view of a monitored region by only visiting *any* m nodes, where $m << n$?" In this context, the protocols *DEEP* [32, 78] and *Supple* [33] were presented. Supple provides an exponential improvement of the number of messages used, when compared to *DEEP*. The choice to use a tree, however, introduces a high overhead of messages transmission to the root and its vicinity. A solution to this consists in creating multiple trees with different roots and load balancing the data dissemination on the different trees, alleviating the communications requirements imposed to a unique root. For this end, we are currently working on the *ProFlex* [46] protocol design. *ProFlex* proposes the use of powerful nodes to create and manage a tree-based distributed storage structure in heterogeneous sensor networks. Instead of using the extra memory features of these nodes, we take profit of their powerful communication range and use the long link to improve data distribution. All the designed and on-going solutions have as main outcome the decoupling of the data dissemination management from the mobile sink's trajectory.

Still in the second chapter, but focusing on information dissemination in wireless social networks, I presented *FairMix* [35, 34, 49] and *VIP delegation* [36, 54] approaches. To improve data dissemination, such works exploit social interests' similarities of people or groups in static networks or the social aspect of their wireless interactions in mobile networks. Interesting open research directions originated from these works, which I am considering in my current on-going research activities and will be discussed in the perspective section hereafter.

Similarly, motivated by the social network properties, the third chapter describes my works on adaptive forwarding services [52] addressing connectivity opportunity in delay tolerant networks. The described protocols, named *Seeker* [41] and *GrAnt* [42, 55], use respectively contact history (contact and communication patterns) and social network properties of nodes to predict future meetings and to better adjust forwarding decisions. The design of such forwarding protocols as well as of *VIP delegation* strategy were based on the observation that social wireless networks display mobility patterns that can be learnt and predicted, and forwarding decisions can be accordingly adjusted. The *GrAnt* design is still an on-going work. Because GrAnt has a lot of flexibility in terms of used metrics and components, we are currently considering different configurations of such elements in order to evaluate their influence in the protocol performance, in terms of Messages Delivery ratio. Finally, *GrAnt* takes profit of the social network properties of nodes and use bi-inspired algorithms to find good path to destinations in the network. In fact, as later mentioned in my perspectives' description, my research activities will continue to be inspired by the adaptation in nature and the features of human social networks.

5.1 Outlooks

(Publications [12]: In collaboration with C. Sengul and A. Ziviani)

Considering practical aspects, all the presented contributions were designed focusing on a particular application scenario, which related requirements were considered in all steps of their design: area monitoring with static or mobile targets; ubiquitous monitoring; health- and wellness-related monitoring; collection of urban-sensing related data in city; distribution of large content to users by service/software providers (e.g., software updates and recurrent security patches); free update of mobile software's ad pools; to cite a few. Moreover, distributed behavior, simplicity, no external intervention, intrinsic and localized knowledge are some of the principles considered by the presented works. Therefore, I believe in their attractiveness and feasibility for an autonomous real deployment.

The only exception in terms of autonomous deployment is the *VIP delegation* approach, which trusts on a collaborative deployment between infrastructure builder and users. In particular, the knowledge concerning the pattern of nodes encounter for a period of one week is required by such approach. For this, we could imagine the network infrastructure builder asking participating users to log their meetings for a certain time. Once collected, such training period would be used by the infrastructure builder to select/upgrade the VIP nodes. From this point, VIPs would act in an autonomous way, disseminating (or collecting) to (or from) any encountered node.

As previously pointed out, the current new emerging factors (i.e, pervasiveness of computing devices, ubiquitous wireless communication, and emergence of new applications and cloud services) emphasize the increasing need for adaptive solutions. Moreover, the new adaptive approaches should consider multiple optimization criteria and variabilities in the system (e.g., both mobility and buffer constraints). I also believe more interdisciplinary approaches will emerge and the WSON research community will continue to be inspired by social structures, biology, games, and control systems.

While adaptive operation will indeed be a necessity, one must take care that the complexity introduced to the system should not outweigh the gains achieved. First, it must be noted that not every functionality should be adaptive. In fact, adaptivity should be applied when it is absolutely necessary. Furthermore, adaptive approaches proposed for different purposes should be able to work together. Therefore, holistic system analysis and evaluation will play an important role to understand the true benefits of adaptive systems for WSONs. I also believe the network protocol design should follow a development cycle, where adaptivity of each component is considered in terms of both complexity and gain, and then, implemented depending on the current constraints. These components should be constantly re-evaluated based on the obtained improvements on network performance and relaxation of constraints in terms of hardware, software, and communication.

In conclusion, though the early solutions to cope with the dynamic nature of WSONs was initiated by DARPA in 1997 [102], the research on adaptive networking services are still increasingly interesting and popular. In particular, solutions that can cope with multiple optimization criteria are increasingly being required, and much work still needs to be done in the area of holistic system analysis/evaluation. I am convinced by the fact that the maturity level of adaptive systems research will rise as adaptive operation becomes a key requirement of protocol design rather than an "afterthought" or "add-on" functionality.

5.2 Future research perspectives

Since 2003, my research activities have been gradually moving from connected self-organizing networks to intermittently connected and opportunistic networks, acknowledging the new communication opportunities and the dynamic shift observed over the past years in wireless networks. In particular, this shift in communication opportunities is the main issue driving my future research activities. Hereafter, I first describe such communication changes and the related motivations of my future research.

Smart portable devices (such as new-generation phones, PDAs or tablet PCs) can be considered as a *pervasive mobile sensing platform* bringing the potential to achieve the pervasive computing vision dreamt by Mark Weiser's in 1991 [10]. Such smart devices are changing the way people are communicating, generating, and exchanging data: (1) They allow the free sensing of data of the surrounding environment anytime and anywhere; (2) They are flexible and have heterogeneous capabilities in terms of type of communication (3G, WiFi, Bluetooth) and of data gathering possibilities (video, audio, image, location, movement, etc). In addition, by their number, smart devices provide an opportunity to gather geo-spatial data with much higher granularity and more penetration than previously possible. This has a very special utility in the developing world, where the deployment of large scale sensor networks was previously cost prohibitive. Finding new ways *to take profit of such "new sensing" opportunities* is essential to bring pervasiveness into the masses.

As a consequence, more and more users have become eager to engage with mobile applications and connected services. Applications related to social networks, global sensing, and content distribution are just a few of the examples. The traffic generated by such smart devices' users has however, caused many problems to 3G network providers, bringing new technical challenges to the networking and telecommunication community: AT&T's subscribers in USA were getting no service or extremely slow because of network straining to meet iPhone users' demand [103]; The company switched from unlimited traffic plans to tiered pricing for 3G data users in summer 2010; Similarly, Dutch T-Mobile's infrastructure has not been able to cope with intense 3G traffic, by thus forcing the company to issue refunds for affected users [104]. Finding new ways *to offload the network and to manage such increasing data usage* is essential to improve the level of service required by the new wave of smart devices' applications.

I believe opportunistic computing and communication provides appealing solutions to the previous mentioned issues, by allowing devices to join forces and leverage heterogeneous resources from other devices. In this context, my future research focus on the following issues:

- I intend to investigate strategies leveraging the uncontrolled mobility patterns of pervasive mobile sensing devices (e.g., smartphones) to improve sensing collaborative efforts: by bridging gaps in existent static sensor networks or by allowing data sensing in areas not previously reachable. This research is based on the consideration that a group of individuals collaborate to define *who, what, and where* to sense and then collectively design a data collection system. For this, two research axes are envisaged:
  - **Sensing recruiting:** Here, opportunistic sensing delegation will be investigated, where through the coordination between pervasive sensing devices, sensing responsibility transfer (or delegation) will be performed. Such delegation will be assisted by social network studies and network resource discovery, which can be accessed via topographic pattern studies and connectivity interactions among participant devices.
  - **Transfer multi-hop coordination:** By taking advantage of users cyclic mobility both in terms of geographic occupancy and interactions, the goal here will be to select the ones to serve as transfer delegates to perform data collection or data distribution in a collaborative fashion. This research activity is an extension of the work performed in *VIP delegation*, where multi-hop delegation and more realistic constraints (as contact duration among users) will be considered.
- One interesting point in any research considering complex network analysis (i.e., a powerful tool to characterize the specific structural features of social networks) is how to perform the mapping from the mobility process generating contacts to the aggregated social graph. Recent studies have shown that the performance of protocols that rely on such complex network analysis heavily depends on how the mapping (contact aggregation) is performed [105]. In fact, the social graph created out of past contacts between nodes should best reflect the underlying mobility structure generating these contacts, so that nodes can be meaningfully differentiated and edges can have predictive value. In the *VIP delegation* work, we have shown that, in both real and synthetic traces, a training period of only 1 week and a threshold-based aggregated social graph yield very good results, i.e., in terms of VIPs' user visits for all the remaining days of the traces. In a similar work related to data

collection delegation, we have also proposed a slotted prediction strategy using a history window of only two days to estimate the likelihood of two producers meeting each other [37, 106]. The obtained results were satisfactory and attested that our prediction strategy, although simple, was also a reliable basis for good delegate set selection. I plan, however, to look deeper into social graph generation techniques from contact traces. This includes evaluating aggregation techniques not only in a time window or contact basis, but also evaluating the moment in time where the network loose its non-trivial topological features, and connectivity patterns become purely random. Additionally, I intend to evaluate in detail the impact of such aggregations in data dissemination strategies to intermittent connected networks.

- The understanding of information dissemination in social mobile opportunistic networks is still a challenging open issue due to the nature of human interaction. I plan to look deeper into what are the factors impacting (in a positive or in a negative way) the success of information dissemination in such type of networks (users' social status, density, visited areas, etc). Additionally, the limited resources of hand-held devices and their dense distribution prevent the use of pure-based flooding techniques, and require the design of more efficient data propagation techniques. In this context, I also intend to investigate the possibility of tailoring network coding for information dissemination in social mobile networks. As shown in the literature, network coding improves throughput and data delivery reliability in wireless networks by better exploiting the characteristics of such networks (such as spatial diversity, broadcast nature of the medium and data redundancy). My aim will be then to define novel strategies to improve dissemination that will encompass both social-aware techniques and network coding-based solutions. The proposed solutions should empower nodes with the ability to make coding decisions adapted to connectivity variations in the network as well as to different content distribution paradigms. In this context, Eduardo Mucelli will start his PhD in October 2011 under my supervision.

Bibliography

[1] A. C. Viana, "Locating and routing in large scale self-organizing networks: From distributed hash tables to adaptive addressing structures," Ph.D. dissertation, Université Pierre et Marie Curie – Sorbonne Universités, Jul. 2005.

[2] A. C. Viana, M. D. Amorim, S. Fdida, and J. F. Rezende, "An underlay strategy for indirect routing," *Springer Wireless Networks*, vol. 10, no. 6, pp. 747–758, Nov. 2004.

[3] ——, "Routage basé sur ancre dans les réseaux à large échelle auto-organisables," in *Colloque Francophone sur l'Ingenierie des Protocoles (CFIP)*, Oct. 2003.

[4] ——, "Routage pair-à-pair dans les réseaux spontanées à large échelle," in *Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications (Algotel)*, May 2003.

[5] ——, "Indirect routing using distributed location information," in *Proc. of IEEE Pervasive Computing and Communication (PerCom) conference*, Mar. 2003.

[6] A. C. Viana, M. D. Amorim, Y. Viniotis, S. Fdida, and J. F. Rezende, "Fairness analysis of the Twins control overhead," CACC, North Carolina State University, Raleigh, USA, Tech. Rep., Jul. 2004.

[7] ——, "Easily-managed location in sons by exploiting space-filling curves," in *Proc. of IEEE Infocom Student Workshop*, Mar. 2005.

[8] ——, "Easily-managed and topological-independent location service for self-organizing networks," in *Proc. of ACM Mobihoc*, May 2005.

[9] ——, "Twins: a dual addressing space representation for self-organizing networks," *IEEE Transactions on Parallel and Distributed Systems (IEEE TPDS)*, vol. 17, no. 12, pp. 1468–1481, Dec. 2006.

[10] M. Weiser, "Hot topics: Ubiquitous computing," *IEEE Computer Communications Magazine*, Oct. 1993.

[11] A. C. Viana, S. Maag, and F. Z. (alphabetical order), "One step forward: Linking wireless self-organising networks validation techniques with formal testing approaches," *ACM Computing Surveys*, vol. 43, no. 2, Jun. 2011.

[12] C. Sengul, A. C. Viana, and A. Ziviani, "A survey of adaptive services to cope with dynamics in wireless self-organizing networks," *To appear in ACM Computing Surveys*, 2012.

[13] A. C. K. Vendramin, A. Munaretto, M. R. Delgado, and A. C. Viana, "Grant: Inferring best forwarders from complex networks' dynamics through a greedy ant colony optimization," *To appear in Elsevier Computer Networks Journal, Special Issue on Complex Dynamic Networks: Tools and Methods*, Mar. 2012.

[14] P. R. W. Junior, M. Fonseca, A. Munaretto, A. C. Viana, and A. Ziviani, "Zap: A distributed channel assignment algorithm for cognitive radio networks," *To appear in Eurasip Journal on Wireless Communications and Networking*, 2011.

[15] T. Gross, P. Steenkiste, and J. Subhlok, "Adaptive distributed applications on heterogenous networks," in *Heterogenous Computing Workshop (HCW)*, April 1999, pp. 209–218.

[16] M. Grossglauser and M. Vetterli, "Locating nodes with ease: Mobility diffusion of last encounters in ad hoc networks," in *Proc. of IEEE Infocom*, Mar. 2003.

[17] F. Benbadis, M. D. de Amorim, and S. Fdida, "Elip: Embedded location information protocol," in *International IFIP-TC6 Networking Conference*, June 2005.

[18] D. Liu, I. Stojmenovic, and X.Jia, "A scalable quorum-based location service in ad hoc and sensor networks," *Intern. Journal of Communication Networks and Distributed Systems*, vol. 1, no. 1, February 2008.

[19] S. Vasudevan, D. Towsley, D. Dennis, and R. Khalili, "Neighbor discovery in wireless networks and the coupon collector's problem," in *Proc. of the ACM Int. Conference on Mobile Computing and Networking (MobiCom)*, Sep. 2009.

[20] C. Drula, C. Amza, F. Rousseau, and A. Duda, “Adaptive energy conserving algorithms for neighbour discovery in opportunistic bluetooth networks,” *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 1, pp. 621 – 655, 2007.

[21] N. Karowski, A. C. Viana, and A. Wolisz, “Optimized asynchronous multi-channel neighbor discovery (5-pages),” in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, Apr. 2011.

[22] A. Willig, N. Karowski, and J.-H. Hauer, “Passive discovery of ieee 802.15.4-based body sensor networks,” *Elsevier Ad Hoc Networks Journal*, vol. 8, no. 7, pp. 742–754, 2010.

[23] “IEEE std 802.15.4-2006,” Sep. 2006.

[24] R. R. Kompella and A. C. Snoeren, “Practical lazy scheduling in sensor networks,” in *Proc. of the International Conference on Embedded Networked Sensor Systems (SenSys)*, Nov. 2003, pp. 280–291.

[25] C. Sengul, A. F. Harris, and R. Kravets, “Reconsidering power-management,” in *4th International Conference on Broadband Communications, Networks, and Systems (Broadnets)*, Sep. 2007, invited paper.

[26] A.-M. Kermarrec, A. Mostéfaoui, M. Raynal, G. Trédan, and A. C. V. (alphabetical order), “Large-scale networked systems: from anarchy to geometric self-structuring,” in *Proc. of 10th International Conference on Distributed Computing and Networking (ICDCN)*, Jan. 2009.

[27] E. L. Merrer, V. Gramoli, A. C. Viana, M. Bertier, and A.-M. Kermarrec, “Energy aware self-organizing density management in wireless sensor networks,” in *Proc. of ACM Mobicom Workshop (ACM MobiShare)*, Sep. 2006.

[28] P. Costa, M. Cesana, S. Brambilla, L. Casartelli, and L. Pizziniaco, “A cooperative approach for topology control in wireless sensor networks: Experimental and simulation analysis,” in *Proc. of the International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2008.

[29] S. M. Rokonuzzaman, R. Pose, and I. Gondal, “A framework for a qos based adaptive topology control system for wireless ad hoc networks with multibeam smart antennas,” in *Proc. of the International Symposium on Parallel and Distributed Processing with Applications (ISPA)*, December 2008.

[30] T. Razafindralambo, N. Mitton, A. C. Viana, M. D. de Amorim, and K. Obraczka, “Adaptive deployment for pervasive data gathering in connectivity-challenged environments,” in *Proc. of IEEE Pervasive Computing and Communication (PerCom) conference*, Mar. 2010.

[31] A. C. Viana and M. D. de Amorim, “Sensing and acting with predefined trajectories,” in *Proc. of ACM Mobihoc Workshop (ACM HeterSanet)*, Hong Kong, China, May 2008.

[32] M. Vecchio, A. C. Viana, A. Ziviani, and R. Friedman, “Deep: Density-based proactive data dissemination protocol for wireless sensor networks with uncontrolled sink mobility,” *Elsevier Computer Communications Journal*, vol. 33, no. 8, pp. 929–939, May 2010.

[33] A. C. Viana, T. Herault, T. Largillier, S. Peyronnet, and F. Zaidi, “Supple: a flexible probabilistic data dissemination protocol for wireless sensor networks,” in *Proc. of ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, Oct. 2010.

[34] G. Karbaschi and A. C. Viana, “A content-based network coding to match social interest similarities in delay tolerant networks,” in *Proc. of ExtremeCom Workshop*, Aug. 2009.

[35] G. Karbaschi, A. C. Viana, S. Martin, and K. A. Agha, “On using network coding in multi-hop wireless networks,” in *Proc. of IEEE PIMRC*, Sep. 2009.

[36] M. V. Barbera, J. Stefa, A. C. Viana, M. D. de Amorim, and M. Boc, “Vip delegation: Enabling vips to offload data in wireless social mobile networks,” in *Proc. of IEEE Intl Conference on Distributed Computing in Sensor Systems (DCOSS)*, Oct. 2011.

[37] G. Biegwood, A. C. Viana, M. D. de Amorim, and M. Boc, “Collaborative data collection in global sensing systems,” in *Proc. of IEEE Conference on. Local Computer Networks (LCN) (4 pages)*, Oct. 2011.

[38] F. Marcelloni and M. Vecchio, “An efficient lossless compression algorithm for tiny nodes of monitoring wireless sensor networks,” *The Computer Journal*, vol. 52, no. 9, November 2009.

[39] M. D. Amorim, F. Benbadis, M. L. Sichitiu, A. C. Viana, and Y. Viniotis, *Routing in Wireless Self-Organizing Networks*. Book chapter at the work Adaptation in Wireless Communications - 2 Volume Set, Taylor and Francis Group, LLC, Aug. 2008, no. ISBN: 9781420045994.

[40] T. Clausen and P. Jacquet, “Optimized link state routing protocol (OLSR),” *RFC 3626*, October 2003.

[41] C. Sengul, A. C. Viana, R. Friedman, M. Bertier, and A.-M. kermarrec, “Adaptive forwarding to match mobility characteristics in delay tolerant networks,” INRIA, Research Report, 2009, RR-6816. [Online]. Available: http://hal.inria.fr/inria-00356601/en/

[42] A. C. K. Vendramin, A. Munaretto, M. R. Delgado, and A. C. Viana, “A greedy ant colony optimization for routing in delay tolerant networks,” in *IEEE Smart Communication Protocols & Algorithms workshop (SCPA) in conjunction with IEEE Globecom 2011*, Dec. 2011.

[43] C. Sengul and R. Kravets, “Heuristic approaches to energy-efficient network design problem,” in *Proc. of the IEEE International Conference on Distributed Computing Systems (ICDCS)*, Jun. 2007.

[44] A.-M. Kermarrec, A. Mostéfaoui, M. Raynal, G. Trédan, and A. C. V. (alphabetical order), “(BA) from anarchy to geometric structuring: the power of virtual coordinates,” in *Proc. of ACM PODC*, Aug. 2008.

[45] A.-M. Kermarrec, A. Mostéfaoui, M. Raynal, G. Trédan, and A. C. Viana, “Large-scale networked systems: from anarchy to geometric self-structuring,” IRISA, Université de Rennes, France, Technical Report, Dec. 2007, PI-1876.

[46] G. Maia, D. L. Guidoni, A. C. Viana, A. L. L. Aquinoz, R. A. F. Mini, and A. F. Loureiro, “Proflex: A probabilistic and flexible data storage protocol for heterogeneous wireless sensor networks,” INRIA, Research Report, 2011, RR-7695. [Online]. Available: http://hal.inria.fr/inria-00610620/en

[47] J. Rahmé, A. C. Viana, and K. A. Agha, “Looking for network functionalities’ extension by avoiding energy-compromised hotspots in wireless sensor networks,” *Springer Annals of Telecommunication Journal (JSAT Special Issue on Home Networking)*, vol. 63, no. 9–10, Oct. 2008.

[48] ——, “Avoiding energy-compromised hotspots in resource-limited wireless networks,” in *Proc. of IFIP 1st Home Networking Conference (IHN)*, Dec. 2007.

[49] G. Karbaschi, A. C. Viana, S. Martin, and K. A. Agha, “On delay fairness for multiple network coding transmissions,” INRIA, Research Report, 2009, RR-6972. [Online]. Available: http://hal.inria.fr/inria-00399561/en/

[50] F. L. Fessant, A. Papadimitriou, A. C. Viana, C. Sengul, and E. Palomar, “A sinkhole resilient protocol for wireless sensor networks: Performance and security analysis,” *To appear in Computer Communication Elsevier Journal*, 2011.

[51] A. C. Viana, A. Ziviani, and R. Friedman, “Decoupling data dissemination from mobile sink’s trajectory in wireless sensor networks,” *IEEE Communications Letters*, vol. 13, no. 3, pp. 178–180, Mar. 2009.

[52] C. Sengul, A. C. Viana, R. Friedman, M. Bertier, and A.-M. kermarrec, “The importance of being adaptive for forwarding,” in *Proc. of ExtremeCom Workshop*, Aug. 2009.

[53] P. R. W. Junior, M. Fonseca, A. Munaretto, A. C. Viana, and A. Ziviani, “Zap: Um algoritmo de atribuição distribuída de canais para mitigação de interferências em redes com rádio cognitivo,” in *Proc. of Brazilian Symposium on Computer Networks and Distributed Systems (SBRC)*, Apr. 2010.

[54] M. V. Barbera, A. C. Viana, M. D. Amorim, and J. Stefa, “Vip delegation: Enabling vips to offload data in wireless social mobile networks,” INRIA, Research Report, 2011, RR-7563. [Online]. Available: http://hal.inria.fr/inria-00573301/en

[55] A. C. K. Vendramin, A. Munaretto, M. R. Delgado, and A. C. Viana, “Grant: Inferring best forwarders from complex networks’ dynamics through a greedy ant colony optimization,” INRIA, Research Report, 2011, RR-7694. [Online]. Available: http://hal.inria.fr/inria-00610558/en

[56] M. H. Rehmani, A. C. Viana, H. Khalife, and S. Fdida, "Surf: A distributed channel selection strategy for data dissemination in multi-hop cognitive radio networks," INRIA, Research Report, 2011, RR-7628. [Online]. Available: http://hal.inria.fr/inria-00596224/en/

[57] G. Trédan and A. C. Viana, "VINCOS – Systémes répartis de grande taille: de l'anarchie à l'auto-structuration," in *Colloque Francophone sur l'Ingénierie des Protocoles (CFIP)*, Les Arcs, France, Mar. 2008.

[58] ——, "Systèmes répartis de grande taille: de l'anarchie à l'auto-structuration," in *Réseaux et Télécoms - Lettre Bimestrielle, Editions Techniques de L'Ingénieur*, May 2008.

[59] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proc. of ACM Mobicom*, Rome, Italie, Jul. 2001, pp. 70–84.

[60] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Optimizing sensor networks in the energy-latency-density design space," *IEEE Transactions on Mobile Computing*, vol. 1, no. 1, pp. 70–80, May 2002.

[61] J. Carle, A. Gallais, D. Simplot-Ryl, and I. Stojmenovic, "Localized sensor area coverage with low communication overhead," in *Proc. of IEEE Percom*, Pisa, Italy, Mar. 2006, pp. 328–337.

[62] M. Dohler, T. Watteyne, D. Barthel, F. Valois, and J.-L. Lu, "Kumar's, zipf's and other laws: How to structure an optimum large-scale wireless (sensor) network?" in *13th European Wireless Conference*, Paris, France, Apr. 2007.

[63] F. Benbadis, K. Obraczka, J. Cortés, and A. Brandwajn, "Exploring landmark placement strategies for topology-based localization in wireless sensor networks," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, 2008.

[64] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic routing without location information," in *Proc. of ACM Mobicom*, Apr. 2003.

[65] A. Caruso, S. Chessa, S. De, and A. Urpi, "Gps free coordinate assignment and routing in wireless sensor netwoks," in *Proc. of IEEE Infocom*, Mar. 2005, pp. 150–160.

[66] B. Liu, P. Brass, O. Dousse, P. Nain, and D. Towsley, "Mobility improves coverage of sensor networks," in *Proc. of ACM MobiHoc*, Urbana-Champaign, IL, USA, May 2005.

[67] D. Hilbert, *Ueber Stetige Abbildung Einer Linie auf ein Flachenstuck.* Mathematische Annalenn, 1891, vol. 38.

[68] B. Moon, H. V. Jagadish, C. Faloutsos, and J. H. Saltz, "Analysis of the clustering properties of the hilbert space-filling curve," *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 1, pp. 124–141, Jan. 2001.

[69] Z. Xu, M. Mahalingam, and M. Karlsson, "Turning heterogeneity into an advantage in overlay routing," in *Proc. of IEEE Infocom*, San Francisco, CA, Mar. 2003.

[70] J. Carle and D. Simplot-Ryl, "Energy-efficient area monitoring for sensor networks," *IEEE Computer*, vol. 37, no. 2, pp. 40–46, 2004.

[71] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. H. Lai, "Deploying wireless sensors to achieve both coverage and connectivity," in *Proc. of ACM MobiHoc*, 2006.

[72] S. Poduri, S. Pattem, B. Krishnamachari, and G. S. Sukhatme, "Using local geometry for tunable topology control in sensor networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 2, pp. 218–230, 2009.

[73] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *ACM International Symposiu on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2004.

[74] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 6, no. 4, pp. 621 – 655, 2008.

[75] K. Huguenin, A.-M. Kermarrec, and E. Fleury, "Route in mobile wsn and get self-deployment for free," in *International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Marina del Rey CA, USA, 2009.

[76] "Wsnet simulator," wsnet.gforge.inria.fr.

[77] E. B. Hamida and G. Chelius, "Strategies for data dissemination to mobile sinks in wireless sensor networks," *IEEE Wireless Communications*, vol. 15, no. 6, pp. 31–37, December 2008.

[78] M. Vecchio, A. C. Viana, A. Ziviani, and R. Friedman, "Proactive data dissemination in wireless sensor networks with uncontrolled sink mobility," INRIA, Research Report, 2009, RR-6820. [Online]. Available: http://hal.inria.fr/inria-00357655/en/

[79] V. Drabkin, R. Friedman, G. Kliot, and M. Segal, "RAPID: Reliable Probabilistic Dissemination in wireless ad-hoc networks," in *Proc. of the 26th IEEE International Symposium on Reliable Distributed Systems (SRDS)*, Oct. 2007.

[80] Z. Bar-Yossef, R. Friedman, and G. Kliot, "RaWMS - random walk based lightweight membership service for wireless ad hoc networks," *ACM Transactions on Computer Systems (TOCS)*, vol. 26, no. 2, pp. 1–66, Jun. 2008.

[81] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TAG: a Tiny AGgregation Service for Ad-Hoc Sensor Networks," *Proc. of ACM SIGOPS Operating Systems Review*, vol. 36, 2002.

[82] J. Eriksson, M. Faloutsos, and S. Krishnamurthy, "Scalable ad hoc routing: The case for dynamic addressing," in *Proc. of IEEE Infocom*, Mar. 2004.

[83] M. H. Rehmani, A. C. Viana, H. Khalife, and S. Fdida, "Improving data dissemination in multi-hop cognitive radio ad-hoc networks," in *Proc. of AdHocNets*, Sep. 2011.

[84] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[85] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," in *Proc. of Conference on Information Sciences and Systems (CISS)*, Aug. 2007, pp. 169–180.

[86] T. H. H. Viswanathan, "Dynamic algorithms for multicast with intra-session network coding," *IEEE Transaction on Information Theory*, vol. 55, no. 2, pp. 797–815, Feb. 2009.

[87] R. W. Yeung, "Multilevel diversity coding with distortion," *IEEE Transaction on Information Theory*, vol. 41, no. 2, p. 412Ű422, Mar. 1995.

[88] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "Xors in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 497–510, June 2008.

[89] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Transaction on Information Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.

[90] "Network coding for wireless mesh networks." [Online]. Available: http://planete.inria.fr/software/NCWM/

[91] M. Gonzalez, C. Hidalgo, and A. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, pp. 779–782, 2008.

[92] D. Kotz, T. Henderson, and I. Abyzov, "CRAWDAD data set dartmouth/campus (v. 2007-02-08)," Downloaded from http://crawdad.cs.dartmouth.edu/dartmouth/campus, Feb. 2007.

[93] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "CRAWDAD data set epfl/mobility (v. 2009-02-24)," Downloaded from http://crawdad.cs.dartmouth.edu/epfl/mobility, Feb. 2009.

[94] A. Mei and J. Stefa, "SWIM: A simple model to generate small mobile worlds," in *Proc. of IEEE Infocom*, 2009.

[95] F. Marcelloni and M. Vecchio, "A simple algorithm for data compression in wireless sensor networks," *IEEE Communications Letters*, vol. 12, no. 6, pp. 411–413, Jun. 2008.

[96] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Duke University, Tech. Rep., 2000.

[97] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Computing Communication Review*, vol. 7, pp. 19–20, July 2003.

[98] A. Balasubramanian, B. N. Levine, and A. Venkaratamani, “DTN routing as a resource allocation problem,” in *ACM SIGCOMM*, Aug. 2007.

[99] N. Eagle and A. S. Pentland, “CRAWDAD data set mit/reality (v. 2005-07-01),” Downloaded from http://crawdad.cs.dartmouth.edu/mit/reality, Jul. 2005.

[100] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, “CRAWDAD data set cambridge/haggle (v. 2009-05-29),” Downloaded from http://crawdad.cs.dartmouth.edu/cambridge/haggle, May 2009.

[101] M. Dorigo, V. Maniezzo, and A. Colorni, “The ant system: Optimization by a colony of cooperating agents,” *IEEE Transactions on Systems, Man, and Cybernetics,Part B: Cybernetics*, vol. 26, no. 1, pp. 29–41, 1996.

[102] M. C. Huebscher and J. A. McCann, “A survey on autonomic computing – degrees, models, and applications,” vol. 40, no. 3, pp. 7–28, August 2008.

[103] “Customers Angered as iPhones Overload AT&T,” New York Times, http://www.nytimes.com/2009/09/03/technology/companies/03att.html, Sep. 2009.

[104] “iPhone overload: Dutch T-Mobile issues refund after 3G issues,” Ars Technica, http://arstechnica.com/tech-policy/news/2010/06/dutch-t-mobile-gives-some-cash-back-because-of-3g-issues.ars, Jul. 2010.

[105] T. Hossmann, T. Spyropoulos, and F. Legendre, “Know thy neighbor: Towards optimal mapping of contacts to social graphs for dtn routing,” in *Proc. of IEEE Infocom*, San Diego, CA, USA, Mar. 2010.

[106] G. Bigwood, A. C. Viana, M. Boc, and M. D. de Amorim, “Opportunistic data collection through delegation,” INRIA, Research Report, 2010, RR-7361. [Online]. Available: http://hal.inria.fr/inria-00508273/en